PROVABLE AND PRACTICAL ALGORITHMS FOR NON-CONVEX PROBLEMS IN MACHINE LEARNING

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Yang Yuan

May 2018

PROVABLE AND PRACTICAL ALGORITHMS FOR NON-CONVEX PROBLEMS IN MACHINE LEARNING

Yang Yuan, Ph.D.

Cornell University 2018

Machine learning has become one of the most exciting research areas in the world, with various applications. However, there exists a noticeable gap between theory and practice. On one hand, a simple algorithm like stochastic gradient descent (SGD) works very well in practice, without satisfactory theoretical explanations. On the other hand, the algorithms analyzed in the theoretical machine learning literature, although with solid guarantees, tend to be less efficient compared with the techniques widely used in practice, which are usually hand tuned or ad hoc based on intuition.

This dissertation is about bridging the gap between theory and practice from two directions. The first direction is "practice to theory", i.e., to explain and analyze the existing algorithms and empirical observations in machine learning. Along this direction, we provide sufficient conditions for SGD to escape saddle points and local minima, as well as SGD dynamics analysis for the two-layer neural network with ReLU activation.

The other direction is "theory to practice", i.e., using theoretical tools to obtain new, better and practical algorithms. Along this direction, we introduce a new algorithm Harmonica that uses Fourier analysis and compressed sensing for tuning hyperparameters. Harmonica supports parallel sampling and works well for tuning neural networks with more than 30 hyperparameters.

BIOGRAPHICAL SKETCH

Yang Yuan was born in Changzhou, China in 1989. In middle school, he learned algorithms and data structures advised by Mr. Wen Cao from Changzhou Senior High School. Afterwards, he did his undergraduate study in computer science at Peking University during 2008-2012, and graduated with highest honor (ranked 2/138) as well as Distinguished Dissertation Award (top 3%, advised by Professor Yao Guo).

He started his Ph.D. in computer science at Cornell under the supervision of Professor Robert D. Kleinberg in 2012. Since then, he worked on various topics in theoretical computer science, including mechanism design, optimization and machine learning. He visited Microsoft Research New England during 2014-2015, and visited Professor Elad Hazan's group at Princeton University in Fall 2016. During his Ph.D., he received Jeff Hawkins & Janet Strauss Fellowship, Amazon AWS Research Award and Microsoft Azure Research Award.

This dissertation is dedicated to my wife Yue Wang, my daughter Eva,
my parents, and my parents-in-law.

ACKNOWLEDGEMENTS

I am extremely fortunate to have Bobby Kleinberg as my advisor, as he gave me the best guidance and support that I could imagine from an advisor. During my PhD, he showed me interesting and approachable theory problems, gave me easy-to-follow guidelines, and taught me beautiful and deep techniques. Moreover, he had huge positive influence on my career path. At the end of my second year, he encouraged me to switch to theoretical machine learning, which turns out to be the tipping point of my research. Afterwards, he took me as a visiting student at MSR New England and sent me to Princeton for one fruitful semester, both of which were crucial to my career development.

More importantly, Bobby has an attractive personality that I really admire. He always tells me that the ultimate goal of research is to satisfy one's intellectual curiosity. He always listens to my thoughts first, then makes decisions or gives suggestions standing at my position rather than his own. When talking to other people, he is always considerate, sincere, responsible and benevolent. Not to mention that he also gave me the critical advice for a crucial event of my family. These are the main reasons that although my PhD journey was long and full of adventure, it was never miserable. Confucius said, *Junzi* treats people in harmony and respects different opinions. Confucius also said, *Ren* is restraining oneself to follow the good morality. This exactly describes my feelings about my advisor Bobby Kleinberg, and is the highest praise that I could think of.

During my PhD, I also got tremendous help from my coauthors. In particular, I would like to thank Rong Ge and Zeyuan Allen-Zhu for their patient guidance. Rong kindly advised me to work on my first project on optimization, which was a great starting point for my later research. Zeyuan coauthored four papers with me, and taught me optimization fundamentals in a clear and intuitive way, with a profound impact on my research tastes and directions. I would like to thank Elad Hazan and Adam Klivans, who are good at attacking problems from refreshing and rigorous theory perspectives, and it was a great fun working with them. I have to thank Nate Foster and Praveen Kumar, for the practical programming skills and principles that I learned by doing a system

project with them, which were completely unexpected but extremely useful for my own research. I would like to thank Gao Huang, who is an expert in deep learning and computer vision, and I learned a lot of practical stuff from him.

I am very fortunate to have Dexter Kozen and Thorsten Joachims in my PhD committee, who have provided kind supports for my academic life. I also want to thank other faculty members for their useful comments and suggestions for my research, in particular Kilian Weinberger, Eva Tardos, Karthik Sridharan, Piotr Indyk, Wei Chen, Jason Hartline, Sanjeev Arora, Siddhartha Banerjee, Yudong Chen and Jon Kleinberg.

Besides the people that I mentioned previously, it was my great pleasure to work with a large group of brilliant and wonderful researchers, including Yuanzhi Li, Chi Jin, Saeed Alaei, Sepideh Mahabadi, Qiantong Xu, Chuan Guo, Yu Sun, Felix Wu, Furong Huang, Manolis Pountourakis, and of course, my dear friend and academic brother Rad Niazadeh.

During my six year long PhD, it was great to have many fantastic friends. In particular, I would like to thank Yizhou Zhang, Yanan Li, Yuxing Tang, Chen Wang, Qin Jia, Yicheng Qin, Weijia Shi, Ge Gao, Dylan Foster, Fu Hu, Tengyu Ma, Xilun Chen, Pooya Jalaly, Rahmtin Rotabi, Samuel Hopkins, Thodoris Lykouris, Surbhi Goel, Ayush Sekhari, Saien Xie and Yongxi Ou.

Finally, I would like to thank my family for their enormous support. My wife Yue makes me a much better person from various aspects, especially makes me less ill-tempered. My 2-year old daughter Eva is a headache, but she is also very cute. I also want to thank my parents for the beliefs and principles they taught me years ago, which have huge influence on every important decision that I made.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Machine learning is a powerful tool with various applications, including image classification, speech recognition, autonomous driving, machine translation, medical images analysis, and many others. Being tightly connected with artificial intelligence and data science, it is entirely conceivable that machine learning will find applications for potentially everything in our daily life, and become one of the driving forces to reshape our future.

However, it is worth noting that there exists a large gap between theory and practice in machine learning. On the practice side, people usually discover efficient methods by trial and error experiments, without provable guarantees of when and why they work well. On the theory side, although people derive rigorous claims for various objects in machine learning, it is hard to apply that knowledge to get new and better algorithms for solving real world problems.

Bridging this gap is both important and rewarding, and can be done with two directions.

One direction is "from practice to theory", where we seek to rigorously explain and analyze the existing algorithms and empirical observations in machine learning. By doing so, we not only build the theoretical foundations for practical algorithms, but also get to understand how different properties of the problems affect the algorithm's performance, which provides theoretical guidance for further improvement.

The other direction is "from theory to practice", where we seek to apply deep and abstract theory tools to obtain new, better and practical algorithms. This direction is exciting and refreshing, because we will get entirely new algorithms, which are usually quite different from the existing ones. Moreover, using well established theory tools, we could easily identify the scenarios in which algorithms will provably work, which is a particularly hard task in machine learning.

This dissertation explores both directions by solving four different problems. The first three problems are along the direction of practice to theory, by explaining and analyzing existing algorithms and empirical observations. More specifically, we show why and when the stochastic gradient descent algorithm, a widely used algorithm in machine learning, escapes saddle points (Chapter 3) and local minima (Chapter 4). We also analyze the dynamics of stochastic gradient descent for training a two layer neural network, which has an intriguing two phase dynamics that can be observed empirically for modern deep networks (Chapter 5). The last problem is along the direction of theory to practice, where we apply Fourier analysis and compressed sensing to get a new practical algorithm for hyperparameter tuning (Chapter 6). See brief introductions for these problems below.

1.1 Escaping from saddle points

Among the numerous optimization methods, stochastic gradient descent (SGD) taught in every introductory machine learning course is undoubtedly the dominant technique being applied in the community. For example, almost all deep neural networks are trained using SGD or its variants.

To optimize a function f, SGD simply runs iterative updates for the weights w_t: $w_{t+1} = w_t - \eta v_t$, where η is the step size[1]. The vector v_t is the stochastic gradient that satisfies $\mathbb{E}[v_t] = \nabla f(w_t)$, and is usually computed using a mini-batch of the dataset.

In the regime of convex optimization, SGD is found to be a nice tradeoff between accuracy and efficiency: it requires more iterations to converge, but fewer gradient evaluations per iteration. For example, for the standard empirical risk minimizing problems with n points and smoothness L, to get to ε-close to w^*[2], gradient descent (GD), which uses the full gradient in every iteration without

[1] In this dissertation, we use step size and learning rate interchangeably.

[2] ε-close means we find a point w such that $\|w - w^*\|_2 \leq \varepsilon$.

any randomness, needs $O(Ln/\varepsilon)$ gradient evaluations [95], but SGD with reduced variance only needs $O(n \log \frac{1}{\varepsilon} + \frac{L}{\varepsilon})$ gradient evaluations [69, 28, 112, 2]. In these scenarios, noise is a by-product of cheap gradient computation, and does not help training.

However, practitioners soon realized that the tradeoff view was insufficient to explain what was happening in practice, especially for non-convex optimization problems like training neural networks. For non-convex problems like neural network training, SGD is not only faster, but also obtains much better solutions compared with GD [71]. Hence, reasonable noise from small mini-batch size seems necessary for successful training.

As one of the first attempts towards understanding this phenomenon, Chapter 3 of this thesis (representing joint work with Rong Ge, Furong Huang, Chi Jin [35]) identified a general property called "strict saddle" for the loss function f, which intuitively means that f has no "flat" saddle points from which no gradient based algorithms could efficiently escape. If f is strict saddle, and the noise in the gradient is non-negligible for every direction, we prove that SGD with appropriate step size will escape all the saddle points and converge to a local minimum within polynomial time. If f also has the property that all local minima are equally good, which holds in many problems [35, 36, 70, 14, 121], our claim indicates that with the help of noise, any point to which SGD finally converges is a global minimum. In other words, SGD provably solves f.

These results underscore how subtle variations in the choice of loss function can affect the time required to solve machine learning problems, and how theory can offer practical guidance for choosing a suitable loss function. For example, for the orthogonal tensor decomposition problem [35], we find that the widely used loss function based on reconstruction error is not strict saddle, which explains why SGD often finds suboptimal solutions. By carefully investigating the Hessian of the loss function, we revise the loss function to make it strict saddle, and observe that the newly designed loss function always outputs much better solutions.

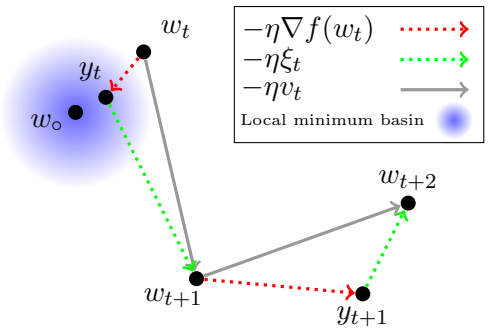


Figure 1.1: SGD path $w_t \to w_{t+1}$ can be decomposed into $w_t \to y_t \to w_{t+1}$. If the local minimum basin has small diameter, the gradient at w_{t+1} will point away from the basin.

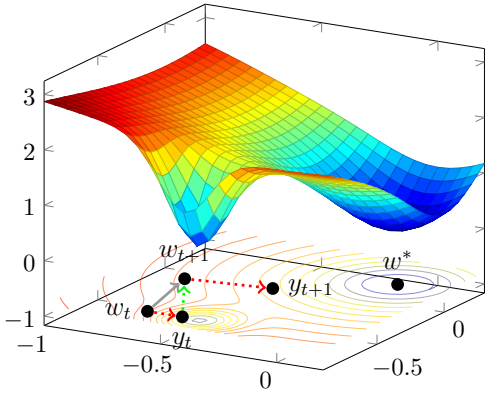


Figure 1.2: 3D version of Figure 1.1: SGD could escape a local minimum within one step.

Subsequent to our work, many other authors have found the strict saddle property to be a powerful tool for proving the correctness of SGD for many machine learning problems with non-convex loss functions, including matrix completion [36], deep linear networks [70], matrix recovery [14], phase retrieval [121], etc. Moreover, follow-up work shows that this property ensures provable guarantees for other algorithms as well. For example, if f is strict saddle, gradient descent converges to a local minimum almost surely with random initialization [79], while normalized gradient descent [80], perturbed gradient descent [68] or accelerated gradient descent [67] can converge to a local minimum more efficiently.

1.2 Escaping from local minima

Another intriguing observation about SGD is that it always converges to "flat" local minima under the correct setting of step sizes [21, 71]. More specifically, if we run SGD with small step sizes, we may get stuck at a sharp local minimum with a bad test error. However, when the step size is initially large and shrinks along the way, it is observed that SGD will escape those bad sharp local minima and finally arrive at a good flat local minimum [58, 85].

See Figure 1.1 for an illustration. Consider the scenario that for some w_t, instead of pointing to the solution w^* (not shown), its negative gradient points to a bad local minimum $w_\circ$, so following the full gradient we will arrive at $y_t \triangleq w_t - \eta\nabla f(w_t)$. Fortunately, since we are running SGD, the actual direction we take is $-\eta v_t = -\eta(\nabla f(w_t) + \xi_t)$, where ξ_t is the noise with $\mathbb{E}[\xi_t] = 0, \xi_t \sim \mathcal{X}(w_t)$, and $\mathcal{X}(w_t)$ is data dependent. As we show in Figure 1.1, if we take a large η, we may get out of the basin region with the help of noise, i.e., from y_t to w_{t+1}. Here, getting out of the basin means the negative gradient at w_{t+1} no longer points to $w_\circ$ (see also Figure 1.2).

To formalize this intuition, in Chapter 4 (representing joint work with Robert Kleinberg and Yuanzhi Li [74]), we look at the sequence $y_t \to y_{t+1}$, where y_t is defined to be $w_t - \eta\nabla f(w_t)$, as in the preceding paragraph. The SGD algorithm never computes these vectors y_t, but we are only using them as an analysis tool. From the equation $w_{t+1} = y_t - \eta\xi_t$ we obtain the following update rule relating y_{t+1} to y_t.

$$y_{t+1} = y_t - \eta\xi_t - \eta\nabla f(y_t - \eta\xi_t) \tag{1.1}$$

The random vector $\eta\xi_t$ in (1.1) has expectation 0, so if we take the expectation of both sides of (1.1), we get $\mathbb{E}_{\xi_t}[y_{t+1}] = y_t - \eta\nabla\mathbb{E}_{\xi_t}[f(y_t - \eta\xi_t)]$. Therefore, if we define $\hat{f}_t$ to be the function $\hat{f}_t(y) = \mathbb{E}_{\xi_t}[f(y - \eta\xi_t)]$, which is simply the original function f convolved with the η-scaled gradient noise, then the sequence y_t is approximately doing gradient descent on the sequence of functions $(\hat{f}_t)$.

This alternative view helps to explain why SGD converges to a good local minimum, even when f has many other sharp local minima. Intuitively, sharp local minima are eliminated by the convolution operator that transforms f to $\hat{f}$, since convolution has the effect of smoothing out short-range fluctuations. This reasoning ensures that SGD converges to a good local minimum under much weaker conditions, because instead of imposing convexity or one-point convexity requirements on f itself, we only require those properties to hold for the smoothed functions $\hat{f}$.

1.3 SGD dynamics for two layer neural network

Nowadays, deep learning is arguably the most powerful technique in machine learning. By learning millions of parameters under very similar architectures, it could achieve state-of-the-art performance on various real world problems.

From an optimization perspective, this is a miracle. It is already hard to believe that a simple three-line algorithm like SGD is enough for training such complicated networks end to end, not to mention that even with different random initializations, SGD would converge to different answers with almost equally good performance.

Unfortunately, we have no rigorous explanations for these observations right now. While the strict saddle property is a handy tool for such problems, it is hard to verify this property for general neural networks due to their highly non-convex nature. Therefore, the existing theoretical results either analyze other algorithms different from SGD [66, 131, 113, 39, 133, 40], or need additional assumptions to simplify the model [4, 5, 22, 70, 18, 111, 70, 45].

Chapter 5 (representing joint work with Yuanzhi Li [82]) makes progress on understanding this mystery by providing a convergence analysis for SGD on a rich subset of two-layer feedforward networks with ReLU activations. This subset is characterized by a special structure called "identity mapping", the most important gadget of the widely used Residual Network [51]. We prove that, if the input follows a Gaussian distribution, with standard $O(1/\sqrt{d})$ initialization of the weights, SGD converges to the global minimum in polynomial time. Identity mapping is necessary for our convergence guarantee because it makes the network asymmetric and thus the global minimum is unique.

Our result differs from traditional non-convex analysis in the sense that we prove global convergence of SGD even when the gradient vectors point away from the global minimum during the

initial training iterations. Our convergence analysis has two "phases" controlled by a potential function g, which represents the distance in columnwise ℓ_2 norm between our current position and the global minimum. In phase I, when g is large, the gradient may point to the wrong direction, so SGD may stray away from the global minimum, but g is guaranteed to decrease. When g becomes sufficiently small, phase II starts, which means SGD enters a nice approximately convex region and easily converges.

This two phase analysis is particularly interesting since with real world data sets (like Cifar-10, Cifar-100) and deep modern networks (like residual network or dense network [57]), empirically we could observe the same "first may move away, then keep getting closer" SGD dynamics [74]. Therefore, our result could be the first step towards understanding the optimization for deep neural networks.

1.4 Sparse learning for hyperparameter tuning

While SGD is a powerful algorithm that works well for most machine learning problems, there are a few notable exceptions, e.g. hyperparameter tuning. Consider the task of training a deep neural network, where one needs to set lots of hyperparameters like the architecture and depth of the network, step size and momentum rate of optimization algorithms, dropout rate, etc. Every possible configuration of these parameters can be seen as an input to a black box function f, while the corresponding network performance is the output. The goal of hyperparameter tuning is to minimize the function f, where querying the value of f is usually expensive. SGD could not be applied here because the parameters are usually discrete and thus gradients are ill-defined.

If f is just random noise, no algorithm can do better than the random search algorithm. Thus, the existing hyperparameter tuning algorithms implicitly assume that f satisfies different assumptions

[117, 123, 118, 34, 126, 65, 81]. However, all such assumptions suffer from the curse of dimensionality: when there are more than 20 hyperparameters, the existing algorithms usually need much more samples than what we could afford, which is why people are still using graduate students to tune hyperparameters.

To solve this problem, Chapter 6 (representing joint work with Elad Hazan and Adam Klivans) proposes a new algorithm called Harmonica [49], which assumes that even in the high dimensional cases, f can be approximated by a small decision tree that maps hyperparameters to function values. We assume that all the hyperparameters are Boolean variables, because we can always discretize continuous variables or binarize categorical variables. This decision tree assumption approximately holds in most deep learning scenarios, in the sense that once we fix the most important hyperparameters, the others only have small incremental contributions to the function value.

Using results from discrete Fourier analysis, we know that any small decision tree T can be approximated by a sparse low degree polynomial h under the Fourier basis of Boolean variables. Therefore, queries for f can be regarded as noisy measurements for the approximating polynomial h. In order to learn f, it suffices to learn h with the noisy measurements.

Applying a deep result from compressed sensing [103], we prove that by running Harmonica with uniform sampling for f, the sparse low degree polynomial h can be recovered with sample complexity linear in the sparsity of h, which further indicates that the decision tree T can be recovered with the same sample complexity as well. Since only uniform sampling is required, Harmonica admits an efficient parallel implementation for the sampling stage, which is the main bottleneck of the hyperparameter tuning problem. Based on simulation results on neural network training, Harmonica (without parallelization) is at least an order of magnitude faster than the state-of-the-art algorithms, and could find results slightly better than what is attainable by hand-tuning.

CHAPTER 2

BACKGROUND MATERIALS

In this chapter we introduce a few notions that will be useful in multiple chapters.

2.1 Notations

Throughout the paper, we use $[d]$ to denote the set $\{1, 2, ..., d\}$. We use bold variables like $\mathbf{W}, \mathbf{I}$ to represent matrices. We use $\|\cdot\|$ to denote the ℓ_2 norm of vectors and spectral norm of matrices, $\|\cdot\|_F$ to denote the Frobenius norm of matrices. For a matrix we use λ_{min} to denote its smallest eigenvalue. For a function $f : \mathbb{R}^d \to \mathbb{R}$, ∇f and $\nabla^2 f$ denote its gradient vector and Hessian matrix.

2.2 Stochastic gradient descent

In general, the stochastic gradient descent algorithm aims to minimize an arbitrary loss function f, with a stochastic gradient oracle SG:

Definition 2.2.1 (Stochastic gradient oracle)**.** *For a function $f(w) : \mathbb{R}^d \to \mathbb{R}$, a function $SG(w)$ that maps a variable to a random vector in $\mathbb{R}^d$ is a Q-bounded stochastic gradient oracle if $\mathbb{E}[SG(w)] = \nabla f(w)$ and $\|SG(w) - \nabla f(w)\| \le Q$.*

Therefore, defining $v_t \triangleq SG(w_t)$, we get the update rule of SGD: $w_{t+1} = w_t - \eta v_t$. If SG is defined carefully with reduced variance, one could get a better convergence guarantee for convex functions [112, 28, 69], which we omit here.

2.3 Smoothness and convexity

We first define a few higher order Lipschitz conditions for a function f.

Definition 2.3.1 (Smoothness)**.** *A function f is L-smooth if for any two points w_1, w_2,*

$$\|\nabla f(w_1) - \nabla f(w_2)\| \leq L\|w_1 - w_2\|. \tag{2.1}$$

Smoothness essentially says f is Lipschitz on gradients. When f is twice differentiable this is equivalent to assuming that the spectral norm of the Hessian matrix is bounded by L.

Definition 2.3.2 (Hessian Smoothness)**.** *A function $f(w)$ is ρ-Hessian smooth, if for any two points w_1, w_2 we have*

$$\|\nabla^2 f(w_1) - \nabla^2 f(w_2)\| \leq \rho\|w_1 - w_2\|. \tag{2.2}$$

Hessian smoothness is a third order condition that is true if the third order derivative exists and is bounded. Below we define a few notions related to convexity.

Definition 2.3.3 (Convexity)**.** *We say a twice-differentiable function is convex if the Hessian at any point is positive semi-definite.*

Definition 2.3.4 (Strong convexity)**.** *We say a twice-differentiable function is λ-strongly convex if the Hessian at any point has smallest eigenvalue at least λ, i.e., $\lambda_{min}(\nabla^2 f(w)) \geq \lambda$.*

In order to get convergence guarantees, sometimes we only need much weaker conditions than convexity, e.g., one point convexity.

Definition 2.3.5 (One point strongly convex)**.** *A function $f(w)$ is called δ-one point strongly convex in domain $\mathbb{D}$ with respect to point w^*, if $\forall w \in \mathbb{D}, \langle -\nabla f(w), w^* - w\rangle > \delta\|w^* - w\|_2^2$.*

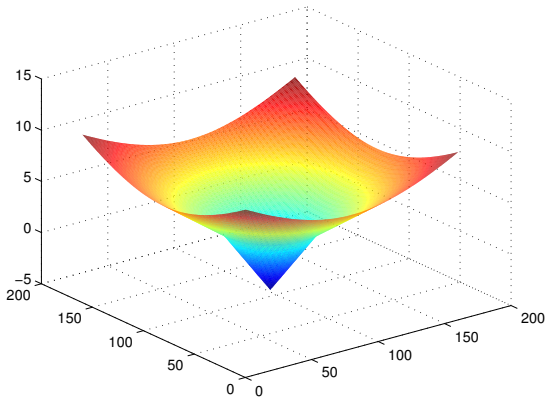


Figure 2.1: The function is one point strongly convex as every point's negative gradient points to the center, but not convex as any line between the center and the red region is below surface.

By definition, if a function f is strongly convex, it is also one point strongly convex in the entire space with respect to the global minimum. However, the reverse is not necessarily true, e.g., see Figure 2.1. If a function is one point strongly convex, then in every step a positive fraction of the negative gradient is pointing to the optimal point. As long as the step size is small enough, we will finally arrive at the optimal point, possibly by a winding path. Formally, we have the following lemma.

Lemma 2.3.6. *For function $f(w)$, consider the SGD update $w_{t+1} = w_t - \eta v_t$, where $\mathbb{E}[v_t|w_t] = \nabla f(w_t)$, $\mathbb{E}[\|v_t|w_t\|^2] \leq V^2$. Suppose for all t, w_t is always inside the δ-one point strongly convex domain with diameter D, i.e., $\|w_t - w^*\| \leq D$. Then for any $\alpha > 0$ and any T such that $T^\alpha \log T \geq \frac{D^2\delta^2}{(1+\alpha)V^2}$ and $\frac{(1+\alpha)\log T}{T} \leq 1$, if $\eta = \frac{(1+\alpha)\log T}{\delta T}$, we have $\mathbb{E}\|w_T - w^*\|^2 \leq \frac{2(1+\alpha)\log T V^2}{\delta^2 T}$.*

Proof. By the updating rule, we have

$$\mathbb{E}\|w_{t+1} - w^*\|^2 = \mathbb{E}\|w_t - w^* - \eta v_t\|^2 = \mathbb{E}\|w_t - w^*\|^2 - 2\langle w_t - w^*, \eta\nabla f(w_t)\rangle + \eta^2\mathbb{E}\|v_t|w_t\|^2$$

$$\leq \mathbb{E}\|w_t - w^*\|^2 - 2\langle w_t - w^*, \eta\nabla f(w_t)\rangle + \eta_t^2 V^2 \leq (1 - 2\eta\delta)\mathbb{E}\|w_t - w^*\|^2 + \eta^2 V^2 \tag{2.3}$$

Now if $\eta\delta\mathbb{E}\|w_t - w^*\|^2 \geq \eta^2 V^2$, we know the $\mathbb{E}\|w_t - w^*\|^2$ will decrease by a factor of $(1 - \eta\delta)$

for every step. Otherwise, although it could increase, we know

$$\mathbb{E}\|w_t - w^*\|^2 \leq \frac{\eta V^2}{\delta}$$

By setting $\eta = \frac{(1+\alpha)\log T}{\delta T}$, we know after T steps, either $\mathbb{E}\|w_T - w^*\|^2$ is already smaller than $\frac{\eta V^2}{\delta} = \frac{(1+\alpha)\log T V^2}{\delta^2 T}$, or it is decreasing by factor of $(1-\eta\delta)$ for every step, which means

$$\mathbb{E}\|w_T - w^*\|^2 \leq \mathbb{E}\|w_0 - w^*\|^2(1-\eta\delta)^T \leq D^2 e^{-\eta\delta T} = D^2 e^{-(1+\alpha)\log T} = \frac{D^2 T^{-\alpha}}{T} \leq \frac{(1+\alpha)\log T V^2}{\delta^2 T}.$$

The last inequality holds since

$$T^\alpha \log T \geq \frac{D^2\delta^2}{(1+\alpha)V^2}$$

Thus, $\mathbb{E}\|w_T - w^*\|^2$ will be smaller than $\frac{(1+\alpha)\log T V^2}{\delta^2 T}$ among the T steps. By the updating rule (2.3), we know that once it is smaller than $\frac{(1+\alpha)\log T V^2}{\delta^2 T}$, after every step it could be at most as large as $\frac{(1+\alpha)\log T V^2}{\delta^2 T} + \eta^2 V^2$, and then it will decrease again. Since $\frac{(1+\alpha)\log T}{T} \leq 1$, the lemma follows. □

Lemma 2.3.6 uses fixed step size, so it easily fits the standard practical scheme that shrinks η by a factor of 10 after every few epochs. For example, we may apply Lemma 2.3.6 every time η gets changed. Notice that our lemma does not imply that w_T will converge to w^*. Instead, it only says w_T will be sufficiently close to w^* with small step size η.

CHAPTER 3

ESCAPING FROM SADDLE POINTS

3.1 Introduction

In this chapter we investigate why stochastic gradient methods can be effective even in presence of saddle points, in particular we answer the following question:

Question: Given a non-convex function f with many saddle points, what properties of f will guarantee stochastic gradient descent to converge to a local minimum efficiently?

We identify a property of non-convex functions which we call *strict saddle*. For strict saddle functions, we show that with only first order (gradient) information, SGD can escape the saddle points efficiently. We give a framework for analyzing SGD in both the unconstrained and equality-constrained cases using this property.

We apply our framework to *orthogonal tensor decomposition*, which is a core problem in learning many latent variable models (see discussion in Subsection 3.1.3). The tensor decomposition problem is inherently susceptible to saddle point issues, as the problem asks to find d different components and any permutation of the true components yields a valid solution. Such symmetry creates exponentially many local minima and saddle points in the optimization problem. Using our new analysis of SGD, we give the first online algorithm for orthogonal tensor decomposition with global convergence guarantee. This is a key step towards making tensor decomposition algorithms more scalable.

3.1.1 Summary of results

Strict saddle functions Given a function $f(w)$ that is twice differentiable, we call w a stationary point if $\nabla f(w) = 0$. A stationary point can either be a local minimum, a local maximum or a saddle point. We identify an interesting class of non-convex functions which we call strict saddle. For these functions the Hessian of every saddle point has a negative eigenvalue. In particular, this means that local second-order algorithms which are similar to the ones in [27] can always make some progress.

It may seem counter-intuitive why SGD can work in these cases: in particular if we run the basic gradient descent starting from a stationary point then it will not move. However, we show that the saddle points are not stable and that the randomness in SGD helps the algorithm to escape from the saddle points.

Theorem 3.1.1 (informal)**.** *Suppose $f(w)$ is strict saddle (see Definition 3.2.2), Noisy Gradient Descent (Algorithm 1) outputs a point that is close to a local minimum in a polynomial number of steps.*

Online tensor decomposition Requiring all saddle points to have a negative eigenvalue may seem strong, but it already allows non-trivial applications to natural non-convex optimization problems. As an example, we consider the orthogonal tensor decomposition problem. This problem is the key step in spectral learning for many latent variable models (see more discussions in Section 3.1.3). Moreover, as we mentioned in Section 1.1, follow up papers identified more strict saddle functions for various machine learning problems as well.

We design a new objective function for tensor decomposition that is strict saddle.

Theorem 3.1.2. *Given random samples X such that $T = \mathbb{E}[g(X)] \in \mathbb{R}^{d^4}$ is an orthogonal 4-th order tensor (see Section 3.1.3), there is an objective function $f(w), w \in \mathbb{R}^{d\times d}$ such that every local*

minimum of $f(w)$ corresponds to a valid decomposition of T. Further, the function f is strict saddle.

Combining this new objective with our framework for analyzing SGD in non-convex settings, we get the first online algorithm for orthogonal tensor decomposition with global convergence guarantee, see Subsection 3.3.

3.1.2 Related work

Relaxed notions of convexity In optimization theory and economics, there are extensive works on understanding functions that behave similarly to convex functions (and in particular can be optimized efficiently). Such notions involve pseudo-convexity [89], quasi-convexity [73], invexity [43] and their variants. More recently there are also works that consider classes that admit more efficient optimization procedures like RSC (restricted strong convexity) [1]. Although these classes involve functions that are non-convex, the function (or at least the function restricted to the region of analysis) still has a unique stationary point that is the desired local/global minimum. Therefore these works cannot be used to prove global convergence for problems like tensor decomposition, where by symmetry of the problem there are multiple local minima and saddle points.

Second-order algorithms The most popular second-order method is Newton's method. Although Newton's method converges fast near a local minimum, its global convergence properties are less understood in the more general case. For non-convex functions, [32] gave a concrete example where a second-order method converges to the desired local minimum in a polynomial number of steps (interestingly the function of interest is trying to find one component in a 4-th order orthogonal tensor, which is a simpler case of our application). As Newton's method often converges also to saddle points, to avoid this behavior, different trust-region algorithms are applied [27].

Stochastic gradient and symmetry The tensor decomposition problem we consider in this paper has the following symmetry: the solution is a set of d vectors $v_1, ..., v_d$. If $(v_1, v_2, ..., v_d)$ is a solution, then for any permutation π and any sign flips $\kappa \in \{\pm 1\}^d$, $(.., \kappa_i v_{\pi(i)}, ...)$ is also a valid solution. In general, symmetry is known to generate saddle points, and variants of gradient descent often perform reasonably in these cases (see [107], [102], [62]). The settings in these works are different from ours, and none of them give bounds on number of steps required for convergence.

There are many other problems that have the same symmetric structure as the tensor decomposition problem, including the sparse coding problem [97] and many deep learning applications [11]. In these problems the goal is to learn multiple "features" where the solution is invariant under permutation. Note that there are many recent papers on iterative/gradient based algorithms for problems related to matrix factorization [64, 111]. These problems often have very different symmetry, as if $Y = AX$ then for any invertible matrix R we know $Y = (AR)(R^{-1}X)$.

Follow up works As mentioned in Section 1.1, there are lots of follow up works after our introduction of the strict saddle property. For example, many authors have found the strict saddle property to be a powerful tool for proving the correctness of SGD for many machine learning problems with non-convex loss functions, including matrix completion [36], deep linear networks [70], matrix recovery [14], phase retrieval [121], etc. Moreover, this property ensures provable guarantees for other algorithms as well. For example, if f is strict saddle, gradient descent converges to a local minimum almost surely with random initialization [79], while normalized gradient descent [80], perturbed gradient descent [68] or accelerated gradient descent [67] can converge to a local minimum more efficiently.

3.1.3 Tensor decomposition

A p-th order tensor is a p-dimensional array of real numbers. In this paper we will mostly consider 4-th order tensors. If $T \in \mathbb{R}^{d^4}$ is a 4-th order tensor, we use $T_{i_1,i_2,i_3,i_4}(i_1, ..., i_4 \in [d])$ to denote its (i_1, i_2, i_3, i_4)-th entry.

Tensors can be constructed from tensor products. We use $(u \otimes v)$ to denote a 2nd order tensor where $(u \otimes v)_{i,j} = u_i v_j$. This generalizes to higher order and we use $u^{\otimes 4}$ to denote the 4-th order tensor

$$[u^{\otimes 4}]_{i_1,i_2,i_3,i_4} = u_{i_1} u_{i_2} u_{i_3} u_{i_4}.$$

We say a 4-th order tensor $T \in \mathbb{R}^{d^4}$ has an *orthogonal decomposition* if it can be written as

$$T = \sum_{i=1}^{d} a_i^{\otimes 4}, \tag{3.1}$$

where a_i's are orthonormal vectors that satisfy $\|a_i\| = 1$ and $a_i^T a_j = 0$ for $i \neq j$. We call the vectors a_i's the components of this decomposition. Such a decomposition is unique up to permutation of a_i's and sign-flips.

A tensor also defines a multilinear form (just as a matrix defines a bilinear form), for a p-th order tensor $T \in \mathbb{R}^{d^p}$ and matrices $M_i \in \mathbb{R}^{d \times n_i} i \in [p]$, we define

$$[T(M_1, M_2, ..., M_p)]_{i_1,i_2,...,i_p} = \sum_{j_1,j_2,...,j_p \in [d]} T_{j_1,j_2,...,j_p} \prod_{t \in [p]} M_t[i_t, j_t].$$

That is, the result of the multilinear form $T(M_1, M_2, ..., M_p)$ is another tensor in $\mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$. We will most often use vectors or identity matrices in the multilinear form. In particular, for a 4-th order tensor $T \in \mathbb{R}^{d^4}$ we know $T(I, u, u, u)$ is a vector and $T(I, I, u, u)$ is a matrix. In particular, if T has the orthogonal decomposition in (3.1), we know $T(I, u, u, u) = \sum_{i=1}^{d} (u^T a_i)^3 a_i$ and $T(I, I, u, u) = \sum_{i=1}^{d} (u^T a_i)^2 a_i a_i^T$.

Given a tensor T with an orthogonal decomposition, the orthogonal tensor decomposition problem asks to find the individual components $a_1, ..., a_d$. This is a central problem in learning many latent variable models, including Hidden Markov Models, multi-view models, topic models, mixtures of Gaussians and Independent Component Analysis (ICA). See the discussion and citations in [3]. The orthogonal tensor decomposition problem can be solved by many algorithms even when the input is a noisy estimation $\tilde{T} \approx T$ [46, 77, 3]. In practice this approach has been successfully applied to ICA [24], topic models [136] and community detection [56].

3.2 Stochastic gradient descent for strict saddle function

In this section we discuss the properties of saddle points, and show if all the saddle points are well-behaved then stochastic gradient descent finds a local minimum for a non-convex function in polynomial time.

3.2.1 Strict saddle property

For a twice differentiable function $f(w)$, we call the points stationary points if their gradients are equal to 0. Stationary points could be local minima, local maxima or saddle points. By local optimality conditions [127], in many cases we can tell what type a point w is by looking at its Hessian: if $\nabla^2 f(w)$ is positive definite then w is a local minimum; if $\nabla^2 f(w)$ is negative definite then w is a local maximum; if $\nabla^2 f(w)$ has both positive and negative eigenvalues then w is a saddle point. These criteria do not cover all the cases as there could be degenerate scenarios: $\nabla^2 f(w)$ can be positive semidefinite with an eigenvalue equal to 0, in which case the point could be a local minimum or a saddle point, or could even be a local maximum, in the case that $\nabla^2 f(w)$ is a zero

matrix.

If a function does not have these degenerate cases, then we say the function is strict saddle:

Definition 3.2.1. *A twice differentiable function $f(w)$ is* strict saddle*, if all of its stationary points satisfy $\lambda_{\min}(\nabla^2 f(w)) \neq 0$.*

Intuitively, if we are not at a stationary point, then we can always follow the gradient and reduce the value of the function. If we are at a saddle point, we need to consider a second order Taylor expansion:

$$f(w + \Delta w) \approx w + (\Delta w)^T \nabla^2 f(w)(\Delta w) + O(\|\Delta w\|^3).$$

Since the strict saddle property guarantees $\nabla^2 f(w)$ to have a negative eigenvalue, there is always a point that is near w and has strictly smaller function value. It is possible to make local improvements as long as we have access to second order information. However it is not clear whether the more efficient stochastic gradient updates can work in this setting.

To make sure the local improvements are significant, we use a robust version of the strict saddle property:

Definition 3.2.2. *A twice differentiable function $f(w)$ is $(\lambda, \gamma, \varepsilon, \delta)$-*strict saddle*, if for any point w at least one of the following is true*

1. $\|\nabla f(w)\| \geq \varepsilon$.
2. $\lambda_{min}(\nabla^2 f(w)) \leq -\gamma$.
3. *There is a local minimum $w^\star$ such that $\|w - w^\star\| \leq \delta$, and the function $f(w')$ restricted to a 2δ-neighborhood of $w^\star$ ($\|w' - w^\star\| \leq 2\delta$) is λ-strongly convex.*

Intuitively, this condition says for any point whose gradient is small, it is either close to a robust local minimum, or is a saddle point (or local maximum) with a significant negative eigenvalue.

Algorithm 1 Noisy Stochastic Gradient

Require: Stochastic gradient oracle $SG(w)$, initial point w_0, desired accuracy κ.
Ensure: w_t that is close to some local minimum $w^\star$.
1: Choose $\eta = \min\{\tilde{O}(\kappa^2/\log(1/\kappa)), \eta_{\max}\}$, $T = \tilde{O}(1/\eta^2)$
2: **for** $t = 0$ to $T - 1$ **do**
3: Sample noise n uniformly from unit sphere.
4: $w_{t+1} \leftarrow w_t - \eta(SG(w) + n)$

We purpose a simple variant of the SGD algorithm, where the only difference to the traditional algorithm is that we add an extra noise term to the updates. The main benefit of this additional noise is that we can guarantee there is noise in every direction, which allows the algorithm to effectively explore the local neighborhood around saddle points. If the noise from the stochastic gradient oracle already has nonnegligible variance in every direction, our analysis also applies without adding additional noise. We show noise can help the algorithm escape from saddle points and optimize strict saddle functions.

Theorem 3.2.3 (Main Theorem)**.** *Suppose a function $f(w) : \mathbb{R}^d \to \mathbb{R}$ that is $(\lambda, \gamma, \varepsilon, \delta)$-strict saddle, and has a stochastic gradient oracle with radius at most Q. Further, suppose the function is bounded by $|f(w)| \le B$, is L-smooth and ρ-Hessian smooth. Then there exists a threshold $\eta_{\max} = \tilde{\Theta}(1)$, so that for any $\zeta > 0$, and for any $\eta \le \eta_{\max}/\max\{1, \log(1/\zeta)\}$, with probability at least $1 - \zeta$ in $t = \tilde{O}(\eta^{-2}\log(1/\zeta))$ iterations, Algorithm 1 (Noisy Gradient Descent) outputs a point w_t that is $\tilde{O}(\sqrt{\eta\log(1/\eta\zeta)})$-close to some local minimum $w^\star$.*

Here (and throughout the rest of the chapter) $\tilde{O}(\cdot)$ ($\tilde{\Omega}, \tilde{\Theta}$) hides the factor that is polynomially dependent on all other parameters (including Q, $1/\lambda$, $1/\gamma$, $1/\varepsilon$, $1/\delta$, B, L, ρ, and d), but independent of η and ζ. So it focuses on the dependency on η and ζ. Our proof technique can give explicit dependencies on these parameters however we hide these dependencies for simplicity of presentation.

Remark 1 (Decreasing learning rate)**.** *Often analysis of stochastic gradient descent uses decreasing*

learning rates and the algorithm converges to a local (or global) minimum. Since the function is strongly convex in the small region close to a local minimum, we can use Theorem 3.2.3 to first find a point that is close to a local minimum, and then apply standard analysis of SGD in the strongly convex case (where we decrease the learning rate by $1/t$ *and get* $1/\sqrt{t}$ *convergence in* $\|w - w^\star\|$*).*

In the next part we sketch the proof of the main theorem. Details are deferred to Appendix A.1.

3.2.2 Proof sketch

In order to prove Theorem 3.2.3, we analyze the three cases in Definition 3.2.2. When the gradient is large, we show the function value decreases in one step (see Lemma 3.2.4); when the point is close to a local minimum, we show with high probability it cannot escape in the next polynomial number of iterations (see Lemma 3.2.5).

Lemma 3.2.4 (Gradient)**.** *Under the assumptions of Theorem 3.2.3, for any point with* $\|\nabla f(w_t)\| \geq C\sqrt{\eta}$ *(where* $C = \tilde{\Theta}(1)$*) and* $C\sqrt{\eta} \leq \varepsilon$*, after one iteration we have* $\mathbb{E}[f(w_{t+1})] \leq f(w_t) - \tilde{\Omega}(\eta^2)$*.*

The proof of this lemma is a simple application of the smoothness property.

Lemma 3.2.5 (Local minimum)**.** *Under the assumptions of Theorem 3.2.3, for any point* w_t *that is* $\tilde{O}(\sqrt{\eta}) < \delta$ *close to local minimum* $w^\star$*, in* $\tilde{O}(\eta^{-2}\log(1/\zeta))$ *number of steps all future* w_{t+i}*'s are* $\tilde{O}(\sqrt{\eta\log(1/\eta\zeta)})$*-close with probability at least* $1 - \zeta/2$*.*

The proof of this lemma is similar to the standard analysis [101] of stochastic gradient descent in the smooth and strongly convex setting, except we only have local strongly convexity. The proof appears in Appendix A.1.

The hardest case is when the point is "close" to a saddle point: it has gradient smaller than ε and smallest eigenvalue of the Hessian bounded by $-\gamma$. In this case we show the noise in our algorithm helps the algorithm to escape:

Lemma 3.2.6 (Saddle point)**.** *Under the assumptions of Theorem 3.2.3, for any point w_t where $\|\nabla f(w_t)\| \leq C\sqrt{\eta}$ (for the same C as in Lemma 3.2.4), and $\lambda_{\min}(\nabla^2 f(w_t)) \leq -\gamma$, there is a number of steps T that depends on w_t such that $\mathbb{E}[f(w_{t+T})] \leq f(w_t) - \tilde{\Omega}(\eta)$. The number of steps T has a fixed upper bound T_{max} that is independent of w_t where $T \leq T_{max} = \tilde{O}(1/\eta)$.*

Intuitively, at point w_t there is a good direction that is hiding in the Hessian. The hope of the algorithm is that the additional (or inherent) noise in the update step makes a small step towards the correct direction, and then the gradient information will reinforce this small perturbation and the future updates will "slide" down the correct direction.

To make this more formal, we consider a coupled sequence of updates $\tilde{w}$ such that the function to minimize is just the local second order approximation

$$\tilde{f}(w) = f(w_t) + \nabla f(w_t)^T (w - w_t) + \frac{1}{2}(w - w_t)^T \nabla^2 f(w_t)(w - w_t).$$

The dynamics of stochastic gradient descent for this quadratic function is easy to analyze as $\tilde{w}_{t+i}$ can be calculated analytically. Indeed, we show the expectation of $\tilde{f}(\tilde{w})$ will decrease. We then use the smoothness of the function to show that as long as the points did not go very far from w_t, the two update sequences $\tilde{w}$ and w will remain close to each other, and thus $\tilde{f}(\tilde{w}_{t+i}) \approx f(w_{t+i})$. Finally we prove the future w_{t+i}'s (in the next T steps) will remain close to w_t with high probability by Martingale bounds. The detailed proof appears in Appendix A.1.

With these three lemmas it is easy to prove the main theorem. Intuitively, as long as there is a small probability of being $\tilde{O}(\sqrt{\eta})$-close to a local minimum, we can always apply Lemma 3.2.4 or

Lemma 3.2.6 to make the expected function value decrease by $\tilde{\Omega}(\eta)$ in at most $\tilde{O}(1/\eta)$ iterations, this cannot go on for more than $\tilde{O}(1/\eta^2)$ iterations because in that case the expected function value will decrease by more than $2B$, but $\max f(x) - \min f(x) \leq 2B$ by our assumption. Therefore in $\tilde{O}(1/\eta^2)$ steps with at least constant probability w_t will become $\tilde{O}(\sqrt{\eta})$-close to a local minimum. By Lemma 3.2.5 we know once it is close it will almost always stay close, so we can repeat this $\log(1/\zeta)$ times to get the high probability result. More details appear in Appendix A.1.

3.2.3 Constrained problems

In many cases, the problem we are facing are constrained optimization problems. In this part we briefly describe how to adapt the analysis to problems with equality constraints (which suffices for the tensor application). Dealing with general inequality constraint is left as future work.

For a constrained optimization problem:

$$\begin{aligned} \min_{w \in \mathbb{R}^d} \quad & f(w) \\ \text{s.t.} \quad & c_i(w) = 0, \qquad i \in [m] \end{aligned} \tag{3.2}$$

in general we need to consider the set of points in a low dimensional manifold that is defined by the constraints. In particular, in the algorithm after every step we need to project back to this manifold (see Algorithm 2 where $\Pi_{\mathcal{W}}$ is the projection to this manifold).

For constrained optimization it is common to consider the Lagrangian:

$$\mathcal{L}(w, \lambda) = f(w) - \sum_{i=1}^{m} \lambda_i c_i(w). \tag{3.3}$$

Under common regularity conditions, it is possible to compute the value of the Lagrangian

Algorithm 2 Projected Noisy Stochastic Gradient

Require: Stochastic gradient oracle $SG(w)$, initial point w_0, desired accuracy κ.
Ensure: w_t that is close to some local minimum $w^\star$.
1: Choose $\eta = \min\{\tilde{O}(\kappa^2/\log(1/\kappa)), \eta_{\max}\}$, $T = \tilde{O}(1/\eta^2)$
2: **for** $t = 0$ to $T - 1$ **do**
3: Sample noise n uniformly from unit sphere.
4: $v_{t+1} \leftarrow w_t - \eta(SG(w) + n)$
5: $w_{t+1} = \Pi_{\mathcal{W}}(v_{t+1})$

multipliers:

$$\lambda^*(w) = \arg\min_{\lambda} \|\nabla_w \mathcal{L}(w, \lambda)\|.$$

We can also define the tangent space, which contains all directions that are orthogonal to all the gradients of the constraints: $\mathcal{T}(w) = \{v : \nabla c_i(w)^T v = 0;\ i = 1, \cdots, m\}$. In this case the corresponding gradient and Hessian we consider are the first-order and second-order partial derivative of Lagrangian $\mathcal{L}$ at point $(w, \lambda^*(w))$:

$$\chi(w) = \nabla_w \mathcal{L}(w, \lambda)|_{(w,\lambda^*(w))} = \nabla f(w) - \sum_{i=1}^{m} \lambda_i^*(w) \nabla c_i(w) \tag{3.4}$$

$$\mathfrak{M}(w) = \nabla^2_{ww} \mathcal{L}(w, \lambda)|_{(w,\lambda^*(w))} = \nabla^2 f(w) - \sum_{i=1}^{m} \lambda_i^*(w) \nabla^2 c_i(w) \tag{3.5}$$

We replace the gradient and Hessian with $\chi(w)$ and $\mathfrak{M}(w)$, and when computing eigenvectors of $\mathfrak{M}(w)$ we focus on its projection on the tangent space. In this way, we can get a similar definition for strict saddle (see Appendix A.2), and the following theorem.

Theorem 3.2.7. *(informal) Under regularity conditions and smoothness conditions, if a constrained optimization problem satisfies strict saddle property, then for a small enough η, in $\tilde{O}(\eta^{-2} \log 1/\zeta)$ iterations Projected Noisy Gradient Descent (Algorithm 2) outputs a point w that is $\tilde{O}(\sqrt{\eta} \log(1/\eta\zeta))$ close to a local minimum with probability at least $1 - \zeta$.*

Detailed discussions and formal version of this theorem are deferred to Appendix A.2.

3.3 Online tensor decomposition

In this section we describe how to apply our stochastic gradient descent analysis to tensor decomposition problems. We first give a new formulation of tensor decomposition as an optimization problem, and show that it satisfies the strict saddle property. Then we explain how to compute stochastic gradient in a simple example of Independent Component Analysis (ICA) [60].

3.3.1 Optimization problem for tensor decomposition

Given a tensor $T \in \mathbb{R}^{d^4}$ that has an orthogonal decomposition

$$T = \sum_{i=1}^{d} a_i^{\otimes 4}, \tag{3.6}$$

where the components a_i's are orthonormal vectors ($\|a_i\| = 1$, $a_i^T a_j = 0$ for $i \neq j$), the goal of orthogonal tensor decomposition is to find the components a_i's.

This problem has inherent symmetry: for any permutation π and any set of $\kappa_i \in \{\pm 1\}, i \in [d]$, we know $u_i = \kappa_i a_{\pi(i)}$ is also a valid solution. This symmetry property makes the natural optimization problems non-convex.

In this section we will give a new formulation of orthogonal tensor decomposition as an optimization problem, and show that this new problem satisfies the strict saddle property.

Previously, [32] solves the problem of finding one component, with the following objective function

$$\max_{\|u\|^2=1} \quad T(u, u, u, u). \tag{3.7}$$

In Appendix A.3.1, as a warm-up example we show this function is indeed strict saddle, and we can apply Theorem 3.2.7 to prove global convergence of stochastic gradient descent algorithm.

It is possible to find all components of a tensor by iteratively finding one component, and do careful *deflation*, as described in [3] or [6]. However, in practice the most popular approaches like Alternating Least Squares [25] or FastICA [59] try to use a single optimization problem to find all the components. Empirically these algorithms are often more robust to noise and model misspecification.

The most straight-forward formulation of the problem aims to minimize the *reconstruction error*

$$\min_{\forall i, \|u_i\|^2=1} \quad \|T - \sum_{i=1}^{d} u_i^{\otimes 4}\|_F^2. \tag{3.8}$$

Here $\|\cdot\|_F$ is the Frobenius norm of the tensor which is equal to the ℓ_2 norm when we view the tensor as a d^4 dimensional vector. However, it is not clear whether this function satisfies the strict saddle property, and empirically stochastic gradient descent is unstable for this objective.

We propose a new objective that aims to minimize the correlation between different components:

$$\min_{\forall i, \|u_i\|^2=1} \quad \sum_{i \neq j} T(u_i, u_i, u_j, u_j), \tag{3.9}$$

To understand this objective intuitively, we first expand vectors u_k in the orthogonal basis formed by $\{a_i\}$'s. That is, we can write $u_k = \sum_{i=1}^{d} z_k(i) a_i$, where $z_k(i)$ are scalars that correspond to the coordinates in the $\{a_i\}$ basis. In this way we can rewrite $T(u_k, u_k, u_l, u_l) = \sum_{i=1}^{d} (z_k(i))^2 (z_l(i))^2$. From this form it is clear that the $T(u_k, u_k, u_l, u_l)$ is always nonnegative, and is equal to 0 only when the support of z_k and z_l do not intersect. For the objective function, we know in order for it to be equal to 0 the z's must have disjoint support. Therefore, we claim that $\{u_k\}, \forall k \in [d]$ is equivalent to $\{a_i\}, \forall i \in [d]$ up to permutation and sign flips when the global minimum (which is 0) is achieved.

We further show that this optimization program satisfies the strict saddle property and all its local minima in fact achieves global minimum value. The proof is deferred to Appendix A.3.2.

Theorem 3.3.1. *The optimization problem (3.9) is $(\lambda, \gamma, \varepsilon, \delta)$-strict saddle, for $\lambda = 1$ and $\gamma, \varepsilon, \delta =$*

$1/poly(d)$. *Moreover, all its local minima have the form* $u_i = \kappa_i a_{\pi(i)}$ *for some* $\kappa_i = \pm 1$ *and permutation* $\pi(i)$.

3.3.2 Implementing stochastic gradient oracle

To design an online algorithm based on objective function (3.9), we need to give an implementation for the stochastic gradient oracle.

In applications, the tensor T is oftentimes the expectation of multilinear operations of samples $g(x)$ over x where x is generated from some distribution $\mathcal{D}$. In other words, for any $x \sim \mathcal{D}$, the tensor is $T = \mathbb{E}[g(x)]$. Using the linearity of the multilinear map, we know $\mathbb{E}[g(x)](u_i, u_i, u_j, u_j) = \mathbb{E}[g(x)(u_i, u_i, u_j, u_j)]$. Therefore we can define the loss function $\phi(u, x) = \sum_{i \neq j} g(x)(u_i, u_i, u_j, u_j)$, and the stochastic gradient oracle $SG(u) = \nabla_u \phi(u, x)$.

For concreteness, we look at a simple ICA example. In the simple setting we consider an unknown signal x that is uniform[1] in $\{\pm 1\}^d$, and an unknown orthonormal linear transformation[2] A ($AA^T = I$). The sample we observe is $y := Ax \in \mathbb{R}^d$. Using standard techniques (see [20]), we know the 4-th order cumulant of the observed sample is a tensor that has orthogonal decomposition. Here for simplicity we don't define 4-th order cumulant, instead we give the result directly.

Define tensor $Z \in \mathbb{R}^{d^4}$ as follows:

$$\begin{aligned} &Z(i,i,i,i) = 3, && \forall i \in [d] \\ &Z(i,i,j,j) = Z(i,j,i,j) = Z(i,j,j,i) = 1, && \forall i \neq j \in [d] \end{aligned}$$

where all other entries of Z are equal to 0. The tensor T can be written as a function of the auxiliary tensor Z and multilinear form of the sample y.

[1] In general ICA the entries of x are independent, non-Gaussian variables.

[2] In general (under-complete) ICA this could be an arbitrary linear transformation, however usually after the "whitening" step (see [20]) the linear transformation becomes orthonormal.

Lemma 3.3.2. *The expectation* $\mathbb{E}[\frac{1}{2}(Z - y^{\otimes 4})] = \sum_{i=1}^{d} a_i^{\otimes 4} = T$*, where* a_i*'s are columns of the unknown orthonormal matrix A.*

This lemma is easy to verify, and is closely related to cumulants [20]. Recall that $\phi(u, y)$ denotes the loss (objective) function evaluated at sample y for point u. Let $\phi(u, y) = \sum_{i \neq j} \frac{1}{2}(Z - y^{\otimes 4})(u_i, u_i, u_j, u_j)$. By Lemma 3.3.2, we know that $\mathbb{E}[\phi(u, y)]$ is equal to the objective function as in Equation (3.9). Therefore we rewrite objective (3.9) as the following stochastic optimization problem

$$\min_{\forall i, \|u_i\|^2 = 1} \quad \mathbb{E}[\phi(u, y)], \text{ where } \phi(u, y) = \sum_{i \neq j} \frac{1}{2}(Z - y^{\otimes 4})(u_i, u_i, u_j, u_j)$$

The stochastic gradient oracle is then

$$\nabla_{u_i} \phi(u, y) = \sum_{j \neq i} \left(\left\langle u_j, u_j \right\rangle u_i + 2 \left\langle u_i, u_j \right\rangle u_j - \left\langle u_j, y \right\rangle^2 \langle u_i, y \rangle \, y \right). \tag{3.10}$$

Notice that computing this stochastic gradient does not require constructing the 4-th order tensor $T - y^{\otimes 4}$. In particular, this stochastic gradient can be computed very efficiently:

Claim 3.3.3. *The stochastic gradient (3.10) can be computed in* $O(d^3)$ *time for one sample or* $O(d^3 + d^2 k)$ *for average of k samples.*

Proof. The proof is straight forward as the first two terms take $O(d^3)$ and is shared by all samples. The third term can be efficiently computed once the inner-products between all the y's and all the u_i's are computed (which takes $O(kd^2)$ time). □

3.4 Experiments

We run simulations for Projected Noisy Gradient Descent (Algorithm 2) applied to orthogonal tensor decomposition. The results show that the algorithm converges from random initial points efficiently

(as predicted by the theorems), and our new formulation (3.9) performs better than reconstruction error (3.8) based formulation.

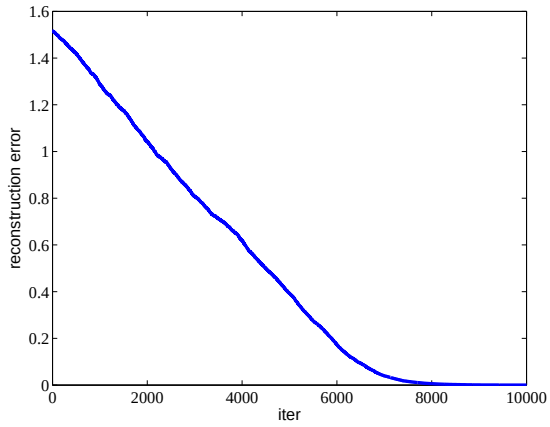
Settings We set dimension $d = 10$, the input tensor T is a random tensor in $\mathbb{R}^{10^4}$ that has orthogonal decomposition (3.1). The step size is chosen carefully for respective objective functions. The performance is measured by normalized reconstruction error $\mathcal{E} = \left(\|T - \sum_{i=1}^{d} u_i^{\otimes 4}\|_F^2\right) / \|T\|_F^2$.

Samples and stochastic gradients We use two ways to generate samples and compute stochastic gradients. In the first case we generate sample x by setting it equivalent to $d^{\frac{1}{4}} a_i$ with probability $1/d$. It is easy to see that $\mathbb{E}[x^{\otimes 4}] = T$. This is a very simple way of generating samples, and we use it as a sanity check for the objective functions.

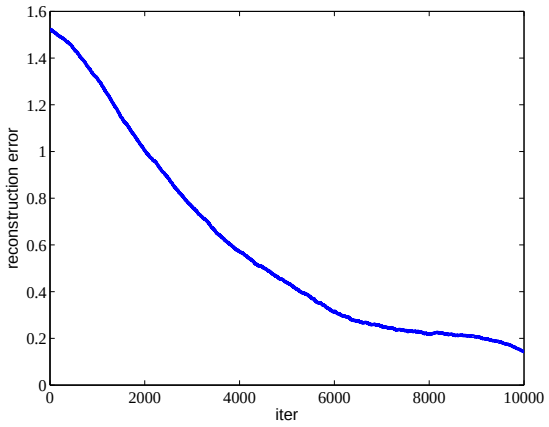
In the second case we consider the ICA example introduced in Section 3.3.2, and use Equation (3.10) to compute a stochastic gradient. In this case the stochastic gradient has a large variance, so we use mini-batch of size 100 to reduce the variance.

Comparison of objective functions We use the simple way of generating samples for our new objective function (3.9) and reconstruction error objective (3.8). The result is shown in Figure 3.1. Our new objective function is empirically more stable (always converges within 10000 iterations); the reconstruction error do not always converge within the same number of iterations and often exhibits long periods with small improvement (which is likely to be caused by saddle points that do not have a significant negative eigenvalue).

Simple ICA example As shown in Figure 3.2, our new algorithm also works in the ICA setting. When the learning rate is constant the error stays at a fixed small value. When we decrease the

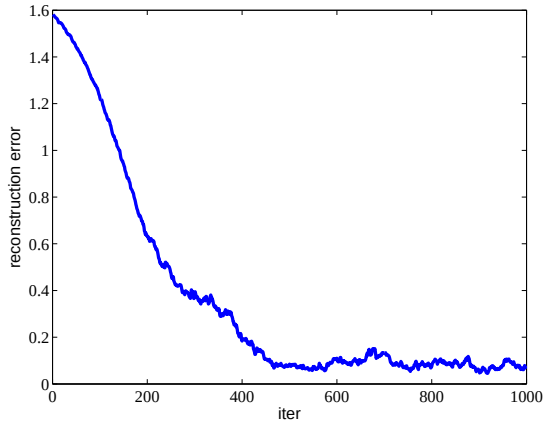


(a) New Objective (3.9)

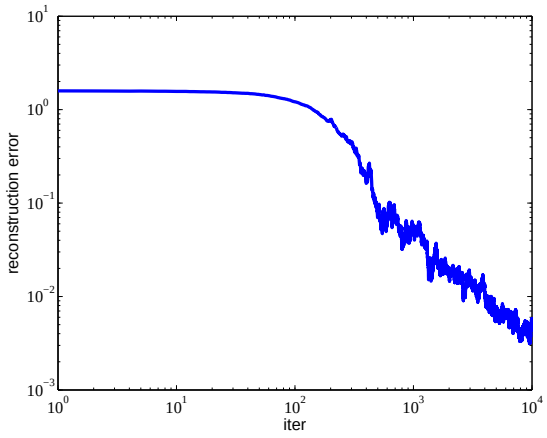


(b) Reconstruction Error Objective (3.8)

Figure 3.1: Comparison of different objective functions



(a) Constant Learning Rate η



(b) Learning Rate η/t (in log scale)

Figure 3.2: ICA setting performance with mini-batch of size 100

learning rate the error converges to 0.

CHAPTER 4

ESCAPING FROM LOCAL MINIMA

4.1 Introduction

In this chapter, we take the alternative view that SGD is essentially acting like GD on the original function f convolved with the gradient noise, see Section 1.2. We can formalize this intuition using the following assumption.

Assumption 4.1.1 (Main Assumption)**.** *For a fixed point w^*, noise distribution $\mathcal{X}(w)$, step size η, the function f is c-one point strongly convex with respect to w^* after convolved with noise. That is, for any w, y in domain $\mathbb{D}$ s.t. $y = w - \eta\nabla f(w)$,*

$$\langle -\nabla \mathbb{E}_{\xi\in\mathcal{X}(w)} f(y - \eta\xi), w^* - y\rangle \geq c\|w^* - y\|_2^2 \tag{4.1}$$

For point y, since the direction $w^* - y$ points to w^*, by having positive inner product with $w^* - y$, we know the direction $-\eta\nabla f(y_t - \eta\xi_t)$ in (1.1) approximately points to w^* in expectation. Therefore, y_t will converge to w^* with decent probability:

Theorem 4.1.2 (Main Theorem, Informal)**.** *Assume f is smooth, for every $w \in \mathbb{D}$, $\mathcal{X}(w)$ s.t., $\max_{\xi\sim\mathcal{X}(w)} \|\xi\|_2 \leq r$. Also assume η is bounded by a constant, and Assumption 4.1.1 holds with w^*, η, and c. For $T_1 \geq \tilde{O}(\frac{1}{\eta c})$[1], and any $T_2 > 0$, with probability at least $1/2$, we have $\|y_t - w^*\|_2^2 \leq O(\log(T_2)\frac{\eta r^2}{c})$ for any t s.t., $T_1 + T_2 \geq t \geq T_1$.*

Notice that our main theorem not only says SGD will get close to w^*, but also says with constant probability, SGD will stay close to w^* for the future T_2 steps. As we will see in Section 4.5, we

[1] We use $\tilde{O}$ to hide log terms here.

observe that Assumption 4.1.1 holds along the SGD trajectory for the modern neural networks when the noise comes from real data mini-batches. Moreover, the SGD trajectory matches with our theory prediction in practice.

Our main theorem can also help explain why SGD could escape "sharp" local minima and converge to "flat" local minima in practice [71]. Indeed, the sharp local minima have small loss value and small diameter, so after convolved with the noise kernel, they easily disappear, which means Assumption 4.1.1 holds. However, flat local minima have large diameter, so they still exists after convolution. In that case, our main theorem says, it is more likely that SGD will converge to flat local minima, instead of sharp local minima.

4.1.1 Related work

Previously, people already realized that the noise in the gradient could help SGD to escape saddle points [35, 68] or achieve better generalization [44, 92]. With the help of noise, SGD can also be viewed as doing approximate Bayesian inference [88]. Besides, it is proved that SGD with extra noise could "hit" a local minimum with small loss value in polynomial time under some assumptions [132]. However, the extra noise is too big to guarantee convergence, and that model cannot deal with escaping sharp local minima.

Escaping sharp local minima for neural network is important, because it is conjectured (although controversial [29]) that flat local minima may lead to better generalization [52, 71, 21]. It is also observed that the correct learning rate schedule (small or large) is crucial for escaping bad local minima [58, 85]. Furthermore, solutions that are farther away from the initialization may lead to wider local minima and better generalization [54]. Under a Bayesian perspective, it is shown that the noise in stochastic gradient could drive SGD away from sharp minima, which decides the optimal

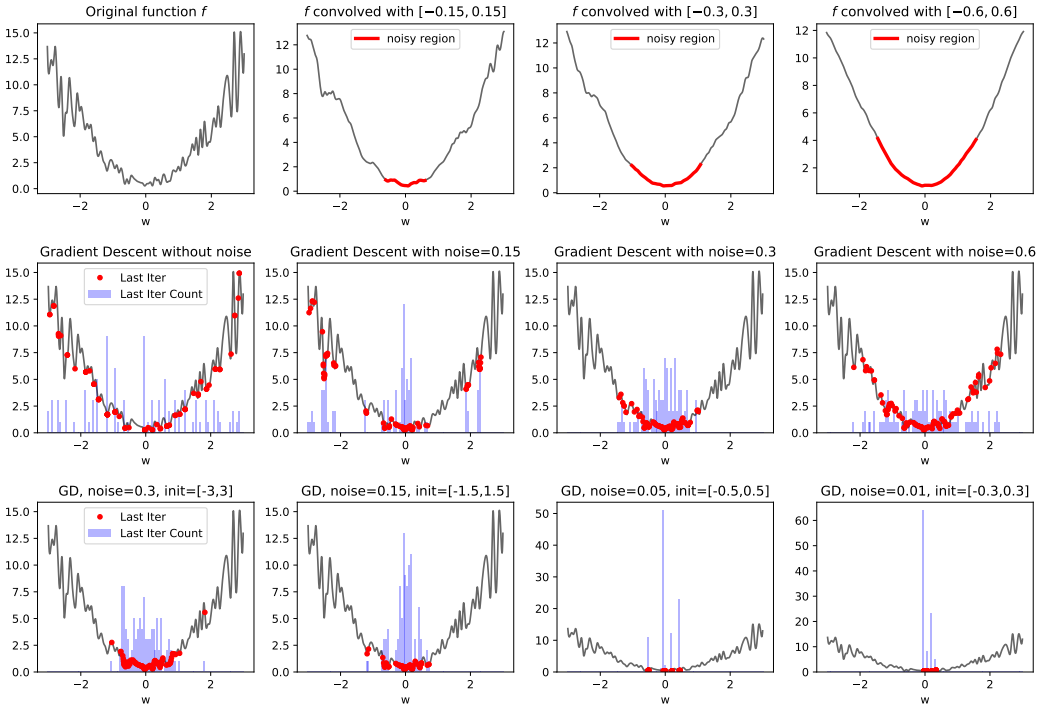


Figure 4.1: Running SGD on a spiky function f. **Row 1:** f gets smoother after convolving with uniform random noise. **Row 2:** Run SGD with different noise levels. Every figure is obtained with 100 trials with different random initializations. Red dots represent the last iterates of these trials, while blue bars represent the cumulative counts. GD without noise easily gets stuck at various local minima, while SGD with appropriate noise level converges to a local region. **Row 3:** In order to get closer to w^*, one may run SGD in multiple stages with shrinking learning rates.

batch size [116]. There are also explanations for why small batch methods prefers flat minima while large batch methods are not, by investigating the canonical quadratic sums problem [100].

To visualize the loss surface of neural network, a common practice is projecting it onto a one dimensional line [42], which was observed to be convex. For the simple two layer neural network, a local one point strongly convexity property provably holds under Gaussian input assumption [82].

4.2 Motivating example

Let us first see a simple example in Figure 4.1. We use $F_{r,c}$ to denote the sub-figure at row r and column c. The function f at $F_{1,1}$ is a approximately convex function, but very spiky. Therefore, GD easily gets stuck at various local minima, see $F_{2,1}$. However, we want to get rid of those spurious local minima, and get a point near $w^* = 0$.

If we take the alternative view that SGD works on the convolved version of f ($F_{1,2}$, $F_{1,3}$, $F_{1,4}$), we find that those functions are much smoother and contain few local minima. However, the gradient noise here is a double-edged sword. On one hand, if the noise is small, the convolved f is still somewhat non-convex, then SGD may find a few bad local minima as shown in $F_{2,2}$. On the other hand, if the noise is too large, the noise dominates the gradient, and SGD will act like random walk, see $F_{2,4}$.

$F_{2,3}$ seems like a nice tradeoff, as all trials converges to a local region near 0, but the region is too big (most points are in $[-1.5, 1.5]$). In order to get closer to 0, we may "restart" SGD with a point in $[-1.5, 1.5]$, using smaller noise level 0.15. Recall in $F_{2,2}$, SGD fails because the convolved f has a few non-convex regions ($F_{1,2}$), so SGD may find spurious local minima. However, those local minima are outside $[-1.5, 1.5]$. The convolved f in $F_{1,2}$ restricted in $[-1.5, 1.5]$ is pretty convex, so if we start a point in this region, SGD converges to a smaller local region centered at 0, see $F_{3,2}$.

We may do this iteratively, with even smaller noise levels and smaller initialization regions, and finally we will get pretty close to 0 with decent probability, see $F_{3,3}$ and $F_{3,4}$.

4.3 Main theorem

Assume that we are running SGD on the sequence $\{w_t\}$. Recall the update rule (1.1) for y_t. Our main theorem says that $\{y_t\}$ is converging to w^* and will stay around w^* afterwards.

Theorem 4.1.2 (Main Theorem)**.** *Assume f is L-smooth, for every $w \in \mathbb{D}$, $\mathcal{X}(w)$ s.t., $\max_{\xi\sim\mathcal{X}(w)} \|\xi\|_2 \le r$. For a fixed target solution w^*, if there exists constant $c, \eta > 0$, such that Assumption 4.1.1 holds with w^*, η, c, and $\eta < \min\{\frac{1}{2L}, \frac{c}{L^2}, \frac{1}{2c}\}$, $\lambda \triangleq 2\eta c - \eta^2 L^2$, $b \triangleq \eta^2 r^2 (1+\eta L)^2$. Then for any fixed $T_1 \ge \frac{\log(\lambda\|y_0 - w^*\|_2^2/b)}{\lambda}$ and $T_2 > 0$, with probability at least $1/2$, we have $\|y_T - w^*\|_2^2 \le \frac{20b}{\lambda}$ and $\|y_t - w^*\|_2^2 \le O\left(\frac{\log(T_2)b}{\lambda}\right)$ for all t s.t., $T_1 + T_2 \ge t \ge T_1$.*

We defer the proof to Section 4.4.

Remark. For fixed c, there exists a lower bound on η to satisfy Assumption 4.1.1, so η cannot be arbitrarily small. However, the main theorem says within $T_1 + T_2$ steps, SGD will stay in a local region centered at w^* with diameter $O\left(\frac{\log(T_2)b}{\lambda}\right)$, which is essentially $\tilde{O}(\eta r^2/c)$ that scales with η. In order to get closer to w^*, a common trick in practice is to restart SGD with smaller step size η' within the local region. If f inside this region has better geometric properties (which is usually true), one gets better convergence guarantee:

Corollary 4.3.1 (Shrinking Learning Rate)**.** *If the assumptions in Theorem 4.1.2 holds, and f restricted in the local region $\mathbb{D}' \triangleq \{w | \|w - w^*\| \le \frac{20b}{\lambda}\}$ satisfy the same assumption with $c' > c, \eta' < \eta$, then if we run SGD with η for the first $T_1 \ge \frac{\log(\frac{\lambda d}{b})}{\lambda}$ steps, and with η' for the next $T_2 \ge \frac{\log(\frac{\lambda}{\lambda'}\frac{20b'}{b'})}{\lambda'}$ steps, with probability at least $1/4$, we have $\|y_{T_1+T_2} - w^*\|_2^2 \le \frac{20b'}{\lambda'} < \frac{20b}{\lambda}$.*

This corollary can be easily generalized to shrink the learning rate multiple times.

Our main theorem is based on the important assumption that the step size is bounded. If the step size is too big, even if the whole function f is one point convex (a stronger assumption than

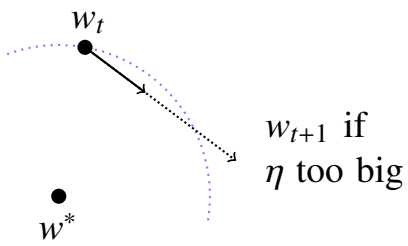


Figure 4.2: When step size is too big, even the gradient is one point convex, we may still go farther away from w^*.

Assumption 4.1.1), and we run full gradient descent, we may not keep getting closer to w^*, as we show below.

Theorem 4.3.2. *For function f, if $\forall w, \langle -\nabla f(x), w^* - w\rangle \le c'\|w^* - w\|_2^2$, and we are at the point w_t. If we run full gradient descent with step size $\eta > \frac{2c'\|w_t - w^*\|_2^2}{\|\nabla f(w_t)\|_2^2}$, we have $\|w_{t+1} - w^*\|_2^2 \ge \|w_t - w^*\|_2^2$.*

Proof. Recall that we have $w_{t+1} = w_t - \eta\nabla f(w_t)$. Since we have $\langle -\nabla f(w_t), w^* - w_t\rangle \le c'\|w^* - w_t\|_2^2$, then

$$\begin{aligned}\|w_{t+1} - w^*\|_2^2 &= \|w_t - \eta\nabla f(w_t) - w^*\|_2^2 \\ &= \|w_t - w^*\|_2^2 + \eta^2\|\nabla f(w_t)\|_2^2 - 2\eta\langle\nabla f(w_t), w_t - w^*\rangle \\ &\ge (1 - 2\eta c')\|w_t - w^*\|_2^2 + \eta^2\|\nabla f(w_t)\|_2^2 > \|w_t - w^*\|_2^2\end{aligned}$$

Where the last inequality holds since we know $\eta > \frac{2c'\|w_t - w^*\|_2^2}{\|\nabla f(w_t)\|_2^2}$. □

This theorem can be best illustrated with Figure 4.2. If η is too big, although the gradient (the arrow) is pointing to the approximately correct direction, w_{t+1} will be farther away from w^* (going outside of the w^*-centered ball).

Although this theorem analyzes the simple full gradient case, SGD is similar. In the high dimensional case, it is natural to assume that most of the noise will be orthogonal to the direction of

$w_t - w^*$, therefore with additional noise inside the stochastic gradient, a large step size will drive w_{t+1} away from w^* more easily.

4.4 Proof for Theorem 4.1.2

In the proof, we will use the following lemma.

Theorem 4.4.1 (Azuma)**.** *Let $X_1, X_2, \cdots, X_n$ be independent random variables satisfying $|X_i - E(X_i)| \leq c_i$, for $1 \leq i \leq n$. We have the following bound for the sum $X = \sum_{i=1}^n X_i$:*

$$\Pr(|X - E(X)| \geq \lambda) \leq 2e^{-\frac{\lambda^2}{2\sum_{i=1}^n c_i^2}}.$$

Our proof has four steps.

Step 1. Since Assumption 4.1.1 holds, we show that SGD always makes progress towards w^* in expectation, plus some noise.

Let filtration $\mathcal{F}_t = \sigma\{\xi_0, \cdots, \xi_{t-1}\}$, where $\sigma\{\cdot\}$ denotes the sigma field. Notice that for any $\xi_t \sim \mathcal{X}(w_t)$, we have $\mathbb{E}[\xi_t|\mathcal{F}_t] = 0$.

Thus,

$$\begin{aligned}
&\mathbb{E}[||y_{t+1} - w^*||_2^2|\mathcal{F}_t] = \mathbb{E}[||y_t - \eta\xi_t - \eta\nabla f(y_t - \eta\xi_t) - w^*||_2^2|\mathcal{F}_t] \\
=&\mathbb{E}\Big[||y_t - \eta\nabla f(y_t - \eta\xi_t) - w^*||_2^2 + ||\eta\xi_t||_2^2 - 2\langle\eta\xi_t, y_t - \eta\nabla f(y_t - \eta\xi_t) - w^*\rangle|\mathcal{F}_t\Big] \\
\leq&\mathbb{E}\Big[||y_t - \eta\nabla f(y_t - \eta\xi_t) - w^*||_2^2 + \eta^2 r^2 - 2\langle\eta\xi_t, -\eta\nabla f(y_t - \eta\xi_t) + \eta\nabla f(y_t) - \eta\nabla f(y_t)\rangle|\mathcal{F}_t\Big] \\
\leq&\mathbb{E}\Big[||y_t - w^*||_2^2 + \eta^2||\nabla f(y_t - \eta\xi_t)||_2^2 - 2\eta\langle-\nabla f(y_t - \eta\xi_t), w^* - y_t\rangle + \eta^2 r^2 + 2\eta^3 r^2 L|\mathcal{F}_t\Big] \\
\leq&||y_t - w^*||_2^2 + \mathbb{E}\Big[\eta^2||\nabla f(y_t - \eta\xi_t)||_2^2|\mathcal{F}_t\Big] + \eta^2 r^2 - 2\eta\langle-\nabla\mathbb{E}_{\xi_t\in\mathcal{X}(w_t)} f(y_t - \eta\xi_t), w^* - y_t\rangle + 2\eta^3 r^2 L \\
\leq&(1 - 2\eta c)||y_t - w^*||_2^2 + \eta^2 r^2 + 2\eta^3 r^2 L + \mathbb{E}\Big[\eta^2 L^2||w^* - y_t + \eta\xi_t||_2^2|\mathcal{F}_t\Big]
\end{aligned}$$

$$\leq (1-2\eta c)\|y_t - w^*\|_2^2 + \eta^2 r^2 + 2\eta^3 r^2 L + \eta^2 L^2 \|w^* - y_t\|_2^2 + \eta^4 r^2 L^2$$
$$= (1 - 2\eta c + \eta^2 L^2)\|y_t - w^*\|_2^2 + \eta^2 r^2 (1+\eta L)^2$$

Step 2. Since SGD makes progress in every step, after many steps, SGD gets very close to w^* in expectation. By Markov inequality, this event holds with large probability.

Notice that since $\eta < \frac{c}{L^2}$, we have $\lambda \triangleq 2\eta c - \eta^2 L^2 > \eta c > 0$. Recall $b \triangleq \eta^2 r^2 (1+\eta L)^2$, we get:

$$\mathbb{E}[\|y_{t+1} - w^*\|_2^2 | \mathcal{F}_t] \leq (1-\lambda)\|y_t - w^*\|_2^2 + b$$

Let $G_t = (1-\lambda)^{-t}(\|y_t - w^*\|_2^2 - \frac{b}{\lambda})$, we get:

$$\mathbb{E}[G_{t+1} | \mathcal{F}_t] \leq G_t$$

That means, G_t is a supermartingale. We have

$$\mathbb{E}[G_{T_1} | \mathcal{F}_{T_1 - 1}] \leq G_0$$

Which gives

$$\mathbb{E}\left[\|y_{T_1} - w^*\|_2^2 - \frac{b}{\lambda} \middle| \mathcal{F}_{T_1-1}\right] \leq (1-\lambda)^{T_1}(\|y_0 - w^*\|_2^2 - \frac{b}{\lambda})$$
$$\leq (1-\lambda)^{T_1}\|y_0 - w^*\|_2^2$$

That is,

$$\mathbb{E}[\|y_{T_1} - w^*\|_2^2 | \mathcal{F}_{T_1-1}] \leq \frac{b}{\lambda} + (1-\lambda)^{T_1}\|y_0 - w^*\|_2^2$$

Since $T_1 \geq \frac{\log\left(\frac{\lambda\|y_0 - w^*\|_2^2}{b}\right)}{\lambda}$, we get:

$$\mathbb{E}[\|y_{T_1} - w^*\|_2^2 | \mathcal{F}_{T_1-1}] \leq \frac{2b}{\lambda}$$

By Markov inequality, we know with probability at least 0.9,

$$\|y_{T_1} - w^*\|_2^2 \leq \frac{20b}{\lambda} \tag{4.2}$$

For notational simplicity, for the analysis below we relabel the point y_{T_1} as y_0. Therefore, at time 0 we already have $\|y_0 - w^*\|_2^2 \leq \frac{20b}{\lambda}$.

Step 3. Conditioned on the event that we are close to w^*, below we show that if for $t_0 > t \geq 0$, y_t is close to w^*, then y_{t_0} is also close to w^* with high probability.

Let $\zeta = \frac{9T_2}{4}$. Let event $\mathfrak{E}_t = \{\forall \tau \leq t, \|y_\tau - w^*\| \leq \mu\sqrt{\frac{b}{\lambda}} = \delta\}$, where μ is a parameter satisfies $\mu \geq \max\{8, 42\log^{\frac{1}{2}}(\zeta)\}$. If with probability $\frac{5}{9}$, $\mathfrak{E}_t$ holds for every $t \leq T_2$, we are done.

By the previous calculation, we know that ($\mathbb{1}_{\mathfrak{E}_t}$ is the indicator function for $\mathfrak{E}_t$)

$$\mathbb{E}[G_t \mathbb{1}_{\mathfrak{E}_{t-1}} | \mathcal{F}_{t-1}] \leq G_{t-1}\mathbb{1}_{\mathfrak{E}_{t-1}} \leq G_{t-1}\mathbb{1}_{\mathfrak{E}_{t-2}}$$

So $G_t \mathbb{1}_{\mathfrak{E}_{t-1}}$ is a supermartingale, with the initial value G_0. In order to apply Azuma inequality, we first bound the following term (notice that we use $\mathbb{E}[\xi_t] = 0$ multiple times):

$$\begin{aligned}
&|G_{t+1}\mathbb{1}_{\mathfrak{E}_t} - \mathbb{E}[G_{t+1}\mathbb{1}_{\mathfrak{E}_t}|\mathcal{F}_t]| \\
=&(1-\lambda)^{-t}|\|y_t - \eta\xi_t - \eta\nabla f(y_t - \eta\xi_t) - w^*\|_2^2 - \mathbb{E}[\|y_t - \eta\xi_t - \eta\nabla f(y_t - \eta\xi_t) - w^*\|_2^2|\mathcal{F}_t]|\mathbb{1}_{\mathfrak{E}_t} \\
\leq&(1-\lambda)^{-t}|2\langle -\eta\xi_t, y_t - \eta\nabla f(y_t - \eta\xi_t) - w^*\rangle + \|\eta\xi_t\|_2^2 + \|y_t - \eta\nabla f(y_t - \eta\xi_t) - w^*\|_2^2 \\
&- \mathbb{E}[2\langle -\eta\xi_t, y_t - \eta\nabla f(y_t - \eta\xi_t) - w^*\rangle + \|\eta\xi_t\|_2^2 + \|y_t - \eta\nabla f(y_t - \eta\xi_t) - w^*\|_2^2|\mathcal{F}_t] \\
=&(1-\lambda)^{-t}|\|\eta\xi_t\|_2^2 - \mathbb{E}[\|\eta\xi_t\|_2^2|\mathcal{F}_t] - 2\langle\eta\xi_t, y_t - \eta\nabla f(y_t - \eta\xi_t) - w^*\rangle + \|y_t - \eta\nabla f(y_t - \eta\xi_t) - w^*\|_2^2 \\
&- \mathbb{E}[2\langle\eta\xi_t, \eta\nabla f(y_t - \eta\xi_t)\rangle + \|y_t - \eta\nabla f(y_t - \eta\xi_t) - w^*\|_2^2|\mathcal{F}_t] \\
\leq&(1-\lambda)^{-t}|\eta^2 r^2 + 2\eta r\|y_t - w^*\| + 2\langle\eta\xi_t, \eta\nabla f(y_t - \eta\xi_t)\rangle + \|\eta\nabla f(y_t - \eta\xi_t) - \eta\nabla f(y_t) + \eta\nabla f(y_t)\|_2^2 \\
&- \mathbb{E}[\|\eta\nabla f(y_t - \eta\xi_t) - \eta\nabla f(y_t) + \eta\nabla f(y_t)\|_2^2|\mathcal{F}_t] + 2\langle y_t - w^*, \eta\nabla f(y_t - \eta\xi_t) \\
&- E[\eta\nabla f(y_t - \eta\xi_t)|\mathcal{F}_t]\rangle - \mathbb{E}[2\langle\eta\xi_t, \eta\nabla f(y_t - \eta\xi_t)\rangle|\mathcal{F}_t] \\
\leq&(1-\lambda)^{-t}|\eta^2 r^2 + 2\eta r\|y_t - w^*\| + 4\eta^2 r\|\nabla f(y_t - \eta\xi_t)\|_2 + \eta^2(2\eta^2 r^2 L^2 + 2\langle\nabla f(y_t), \nabla f(y_t - \eta\xi_t)
\end{aligned}$$

$$
\begin{aligned}
&- \nabla f(y_t) - \mathbb{E}[\nabla f(y_t - \eta\xi_t) - \nabla f(y_t)|\mathcal{F}_t]\rangle) + 2\eta\Big\langle y_t - w^*, \nabla f(y_t - \eta\xi_t) - \nabla f(y_t) \\
&- E[\nabla f(y_t - \eta\xi_t) - \nabla f(y_t)|\mathcal{F}_t]\Big\rangle \\
=&(1-\lambda)^{-t}|\eta^2 r^2 + 2\eta r\|y_t - w^*\| + 4\eta^2 rL(\eta r + \|y_t - w^*\|_2) \\
&+ \eta^2\left(2\eta^2 r^2 L^2 + 4L\|y_t - w^*\|_2 \eta r L\right) + 4\eta^2 rL\|y_t - w^*\| \\
\leq&(1-\lambda)^{-t}\left(3.5\eta^2 r^2 + 7\eta r\delta\right)
\end{aligned}
$$

Where the last inequality uses the fact that $\eta L \leq \frac{1}{2}$ and $\|y_t - w^*\|_2 \leq \delta$ (as $\mathbb{1}_{\mathfrak{E}_t}$ holds). Let $M \triangleq 3.5\eta^2 r^2 + 7\eta r\delta$. Let $d_\tau = |G_\tau \mathbb{1}_{\mathfrak{E}_{\tau-1}} - \mathbb{E}[G_\tau \mathbb{1}_{\mathfrak{E}_{\tau-1}}|\mathcal{F}_t]|$, we have

$$
\sum_{\tau=1}^{t} d_\tau^2 = \sum_{\tau=1}^{t}(1-\lambda)^{-2\tau}M^2
$$

$$
r_t = \sqrt{\sum_{\tau=1}^{t} d_\tau^2} = M\sqrt{\sum_{\tau=1}^{t}(1-\lambda)^{-2\tau}}
$$

Apply Azuma inequality (Theorem 4.4.1), for any $\zeta > 0$, we know

$$
\Pr(G_t \mathbb{1}_{\mathfrak{E}_{t-1}} - G_0 \geq \sqrt{2} r_t \log^{\frac{1}{2}}(\zeta)) \leq \exp\left(\frac{-2r_t^2 \log(\zeta)}{2\sum_{\tau=1}^{t} d_\tau^2}\right) = \exp^{-\log(\zeta)} = \frac{1}{\zeta}
$$

Therefore, with probability $1 - \frac{1}{\zeta}$,

$$
G_t \mathbb{1}_{\mathfrak{E}_{t-1}} \leq G_0 + \sqrt{2} r_t \log^{\frac{1}{2}}(\zeta)
$$

Step 4. The inequality above says, if $\mathfrak{E}_{t-1}$ holds, i.e., for all $\tau \leq t-1, \|y_\tau - w^*\| \leq \delta$, then with probability $1 - \frac{1}{\zeta}$, G_t is bounded. If we can show from the upper bound of G_t that $\|y_t - w^*\| \leq \delta$ is also true, we automatically get $\mathfrak{E}_t$ holds. In other words, that means if $\mathfrak{E}_{t-1}$ holds, then $\mathfrak{E}_t$ holds with probability $1 - \frac{1}{\zeta}$. Therefore, by applying this claim T_2 times, we get $\mathfrak{E}_{T_2}$ holds with probability

$1 - \frac{T_2}{\zeta} = \frac{5}{9}$. Combining with inequality (4.2), we know with probability at least $1/2$, the theorem statement holds. Thus, it remains to show that $\|y_t - w^*\| \leq \delta$.

If $G_t \mathbb{1}_{\mathfrak{E}_{t-1}} \leq G_0 + \sqrt{2} r_t \log^{\frac{1}{2}}(\zeta)$, we know

$$(1-\lambda)^{-t}\left(\|y_t - w^*\|_2^2 - \frac{b}{\lambda}\right) \leq \|y_0 - w^*\|_2^2 - \frac{b}{\lambda} + \sqrt{2} r_t \log^{\frac{1}{2}}(\zeta)$$

So

$$\begin{aligned}\|y_t - w^*\|_2^2 \leq& (1-\lambda)^t \left(\|y_0 - w^*\|_2^2 + \sqrt{2} r_t \log^{\frac{1}{2}}(\zeta)\right) + \frac{b}{\lambda} \\ \leq& \|y_0 - w^*\|_2^2 + \sqrt{2}(1-\lambda)^t r_t \log^{\frac{1}{2}}(\zeta) + \frac{b}{\lambda}\end{aligned}$$

Notice that

$$\begin{aligned}(1-\lambda)^t r_t =& (1-\lambda)^t M \sqrt{\sum_{\tau=1}^{t} (1-\lambda)^{-2\tau}} = M \sqrt{\sum_{\tau=1}^{t} (1-\lambda)^{2(t-\tau)}} \\ =& M \sqrt{\sum_{\tau=0}^{t-1} (1-\lambda)^{2\tau}} \leq M \sqrt{\frac{1}{1-(1-\lambda)^2}} \leq \frac{M}{\sqrt{\eta c}}\end{aligned}$$

The second last inequality holds because we know $\frac{1}{1-(1-\lambda)^2} = \frac{1}{2\lambda - \lambda^2} \leq \frac{1}{\lambda} \leq \frac{1}{\eta c}$, since $\lambda = 2\eta c - \eta^2 L^2 \leq 2\eta c < 1$, and $\lambda > \eta c$.

That means,

$$\begin{aligned}\|y_t - w^*\|_2^2 \leq& \|y_0 - w^*\|_2^2 + \frac{\sqrt{2} M}{\sqrt{\eta c}} \log^{\frac{1}{2}}(\zeta) + \frac{b}{\lambda} \\ \leq& \frac{\sqrt{2}(3.5\eta^2 r^2 + 7\eta r \delta)}{\sqrt{\eta c}} \log^{\frac{1}{2}}(\zeta) + \frac{21 b}{\lambda}\end{aligned}$$

It remains to prove the following lemma.

Lemma 4.4.2.

$$\frac{\sqrt{2}(3.5\eta^2 r^2 + 7\eta r \delta)}{\sqrt{\eta c}} \log^{\frac{1}{2}}(\zeta) + \frac{21 b}{\lambda} \leq \delta^2$$

Proof. Recall that we want to show

$$\frac{\sqrt{2}(3.5\eta^2 r^2 + 7\eta r\delta)}{\sqrt{\eta c}} \log^{\frac{1}{2}}(\zeta) + \frac{21b}{\lambda} \leq \delta^2 = \frac{\mu^2 b}{\lambda} = \frac{\mu^2\eta^2 r^2(1+\eta L)^2}{\lambda}$$

On the left hand side there are three summands. Below we show that each of them is bounded by $\frac{\mu^2 b}{3\lambda}$[2].

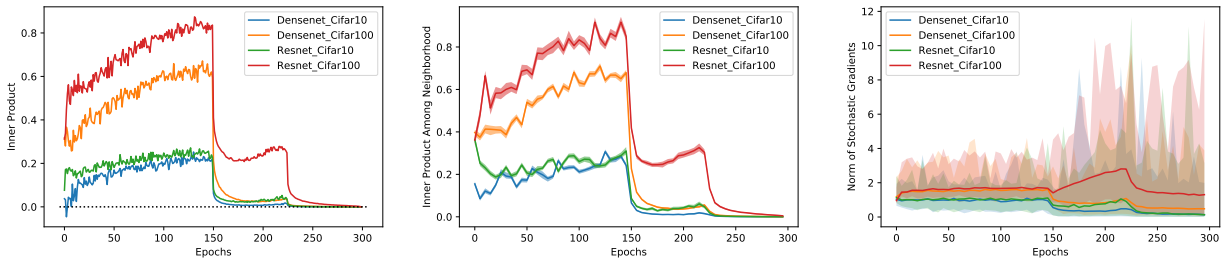
Since $\mu \geq \max\{8, 42\log^{\frac{1}{2}}(\zeta)\}$, we know $\frac{21b}{\lambda} \leq \frac{63b}{3\lambda} < \frac{8^2 b}{3\lambda} \leq \frac{\mu^2 b}{3\lambda}$. Next, we have

$$\begin{aligned}
&42\log^{\frac{1}{2}}(\zeta) \leq \mu \\
\Rightarrow &\sqrt{30\log^{\frac{1}{2}}(\zeta)\eta^{0.5}c^{0.5}} \leq \mu \\
\Rightarrow &15\log^{\frac{1}{2}}(\zeta) \leq \frac{\mu^2}{2\eta^{0.5}c^{0.5}} \\
\Rightarrow &\frac{15}{\sqrt{c}}\log^{\frac{1}{2}}(\zeta) \leq \frac{\mu^2\eta^{0.5}}{\lambda} \\
\Rightarrow &\frac{3.5\sqrt{2}\eta^{1.5}r^2}{\sqrt{c}}\log^{\frac{1}{2}}(\zeta) \leq \frac{\mu^2\eta^2 r^2}{3\lambda} \\
\Rightarrow &\frac{3.5\sqrt{2}\eta^2 r^2}{\sqrt{\eta c}}\log^{\frac{1}{2}}(\zeta) \leq \frac{\mu^2\eta^2 r^2(1+\eta L)^2}{3\lambda}
\end{aligned}$$

Finally,

$$\begin{aligned}
&42\log^{\frac{1}{2}}(\zeta) \leq \mu \\
\Rightarrow &\frac{42}{\sqrt{c}}\log^{\frac{1}{2}}(\zeta) \leq \mu\sqrt{\frac{1}{c}} \\
\Rightarrow &\frac{7\sqrt{2}\eta r}{\sqrt{\eta c}}\log^{\frac{1}{2}}(\zeta) \leq \frac{\mu\sqrt{\frac{\eta^2 r^2(1+\eta L)^2}{2\eta c}}}{3}
\end{aligned}$$

[2]We made no effort to optimize the constants here.



(a) SGD trajectory is locally one point convex.

(b) The neighborhood of SGD trajectory is one point convex.

(c) The norm of stochastic gradient

Figure 4.3: (a). The inner product between the negative gradient and $w_{300} - w_t$ for each epoch $t \geq 5$ is always positive. Every data point is the **minimum** value among 5 trials. (b). Neighborhood of SGD trajectory is also one point convex with respect to w_{300}. (c). Norm of stochastic gradient

$$\Rightarrow \frac{7\sqrt{2}\eta r}{\sqrt{\eta c}} \log^{\frac{1}{2}}(\zeta) \leq \frac{\delta}{3}$$
$$\Rightarrow \frac{7\sqrt{2}\eta r\delta}{\sqrt{\eta c}} \log^{\frac{1}{2}}(\zeta) \leq \frac{\delta^2}{3}$$

Adding the three summands together, we get the claim. □

Therefore, $\|y_t - w^*\| \leq \delta$. Combining the 4 steps together, we have proved the theorem.

4.5 Empirical observations

In this section, we explore the loss surfaces of modern neural networks, and show that they enjoy many nice one point convex properties. Therefore, our main theorem could be used for explaining why SGD works so well in practice.

4.5.1 The SGD trajectory is one point convex

It is well known that the loss surface of neural network is highly non-convex, with numerous local minima. However, we observe that the loss surface is consisted of many one point convex basin region, while each time SGD traverses one of such regions.

See Figure 4.3a for details. We ran experiments on Resnet [51] (34 layers, $\approx$ 1.2M parameters), Densenet [57] (100 layers, $\approx$ 0.8M parameters) on Cifar10 and Cifar100, each for 5 trials with 300 epochs and different initializations. For the start of every epoch x_t in each trial, we compute the inner product between the negative gradient $-\nabla f(w_t)$ and the direction $w_{300} - w_t$. In Figure 4.3a, we plot the minimum value for every epoch among 5 trials for each setting. Notice that except for the starting period of densenet on Cifar-10, all the other networks in all trials have positive inner products, which shows that the trajectory of SGD (except the starting period) is one point convex with respect to the final solution[3]. In these experiments, we have used the standard step size schedule (0.1 initially, 0.01 after epoch 150, and 0.001 after epoch 225). However, we got the same observation when using smoothly decreasing step sizes (shrink by 0.99 per epoch).

4.5.2 The neighborhood of the trajectory is one point convex

Having a one point convex trajectory for 5 trials does not suffice to show SGD always has a simple and easy trajectory, due to the randomness of the stochastic gradient. Indeed, by a slight random perturbation, SGD might be in a completely different trajectory that is far from being one point convex to the final solution. However, in this subsection, we show that it is not the case, as the SGD trajectory is one point convex after convolving with uniform ball with radius 0.5. That means, the whole neighborhood of the SGD trajectory is one point convex with respect to the final solution.

[3]Similar observations were implicitly observed previously [42].

In this experiment, we tried Resnet (34 layers, $\approx$ 1.2M parameters), Densenet (100 layers, $\approx$ 0.8M parameters) on Cifar10 and Cifar100. For every epoch in each setting, we take one point and look at its neighborhood with radius 0.5 (upper bound of the length of one SGD step, as we will show below). We take 100 random points inside each neighborhood to verify Assumption 4.1.1. More specifically, for every random point w in the neighborhood of w_t, we computer $\langle -\nabla f(w), w_{300} - w_t \rangle$. Figure 4.3b shows the mean value (solid line), as well as upper and lower bound of the inner product (shaded area). As we can see, the inner products for all epochs in every setting have small variances, and are always positive. Although we could not verify Assumption 4.1.1 by computing the exact expectation due to limited computational resources, from Figure 4.3b and Hoeffding bound (Lemma 4.5.1), we conclude that Assumption 4.1.1 should hold with high probability.

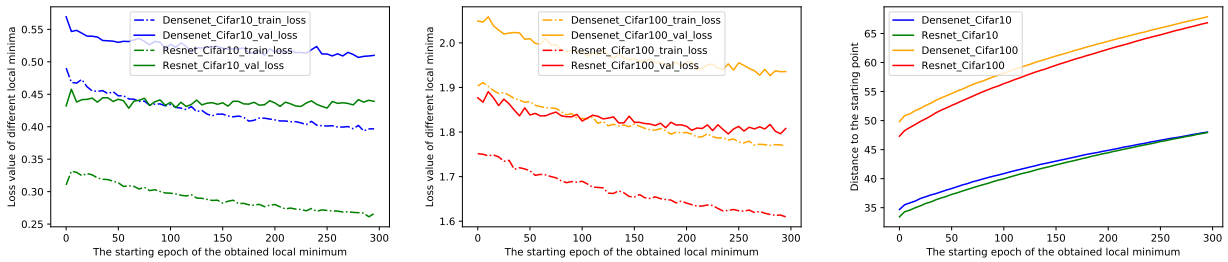
Lemma 4.5.1 (Hoeffding bound [53])**.** *Let $X_1, \ldots, X_n$ be i.i.d. random variables bounded by the interval $[a, b]$. Then* $\Pr\left(\frac{1}{n}\sum_i X_i - \mathbb{E}[X_1] \geq t\right) \leq \exp\left(-\frac{2nt^2}{(b-a)^2}\right)$.

Figure 4.3c shows the norm of the stochastic gradients, including both the mean value (solid lines), as well as upper and lower bounds (shaped area). For all settings, the stochastic gradients are always less than 5 before epoch 150 with learning rate 0.1, and less than 15 afterwards with learning rate 0.01. Therefore, the step size of SGD is always bounded by 0.5.

Notice that the gradient norm gets bigger when we get closer to the final solution (after epoch 150). This further explains why shrinking step size is important.

4.5.3 Loss surface is locally a "slope"

Even with the observation that the whole neighborhood along the SGD trajectory is one point convex with respect to the final solution, there exists a chicken-and-egg concern, as the final target



(a) Loss value of different local minima on Cifar10

(b) Loss value of different local minima on Cifar100

(c) Distance from the local minima to the initialization

Figure 4.4: Spectrum of local minima on the loss surface on modern neural networks.

is generated using the SGD trajectory.

In this subsection, we show that the one point convexity is a pretty "global" property. We were running Resnet and Densenet on Cifar10, but with smaller networks (each with about $10K$ parameters). For each network, if we fix the first 10 epochs, and generate 50 SGD trajectories with different random seeds for 140 epochs and 0.1 learning rate, we get 50 different final solutions (they are pretty far away from each other, with minimum pairwise distance 40). For each network, if we look at the inner product between the negative gradient of **any** epoch of **any** trajectories, and the vector pointing to **any** final solutions, we find that the inner products are almost always positive. (only 0.1% of the inner products are not positive for Densenet, and only 2 out of $343,000$ inner products are not positive for Resnet).

This indicates that the loss surface is "skewed" to the similar direction, and our observation that the whole SGD trajectory is one point convex w.r. to the last point is not a coincidence. Based on our Theorem 4.1.2, such loss surface is very friendly to SGD optimization, even with a few exceptional points that are not one point convex with respect to the final solution.

Notice that in general, it is not possible that all the negative gradients of all points are one point convex with respect to multiple target points. For example, if we take $1D$ interpolation between any two target points, we could easily find points that have negative gradients only pointing to one target

point. However, based on our simulation, empirically SGD almost never traverse those regions.

4.5.4 Spectrum of the local minima

From the previous subsections, we know that the loss surface of neural network has great one point convex properties. It seems that by our Theorem 4.1.2, SGD will almost always converge to a few target points (or regions). However, empirically SGD converges to very different target points. In this subsection, we argue that this is because of the learning rate is too big for SGD to converge (Theorem 4.3.2). On the other hand, whenever we shrink the learning rate to 0.01, Theorem 4.1.2 immediately applies and SGD converges to a local minimum.

In this experiment, we were running smaller version of Resnet and Densenet (each with about $10K$ parameters) on Cifar10 and Cifar100. For each setting, we first train the network with step size 0.1 for 300 epochs, then we pick different epochs as the new starting points for finding nearby local minima using smaller learning rates with additional 150 epochs.

See Figure 4.4a and Figure 4.4b. Starting from different epochs, we got local minima with decreasing validation loss and training loss.

To show that these local minima are not from the same region, we also plot the distance of the local minima to the (unique) initialized point. As we can see, as we pick later epochs as the starting points, we get local minima that are farther away from the initialization with better quality (also observed in [54]).

Furthermore, we observe that for every local minimum, the whole trajectory is always **one point convex** to that local minimum. Therefore, the time for shrinking learning rate decides the quality of the final local minimum. That is, using large step size initially avoids being trapped into a bad local

minimum, and whenever we are distant enough from the initialization, we can shrink the step size and converge to a good local minimum (due to one point convexity by Theorem 4.1.2).

CHAPTER 5

TWO LAYER NETWORK CONVERGENCE ANALYSIS

5.1 Introduction

In this chapter, we give the first convergence analysis of SGD for two-layer feedforward network with ReLU activations. For this basic network, it is known that even in the simplified setting where the weights are initialized symmetrically and the ground truth forms orthonormal basis, gradient descent might get stuck at saddle points [124]. Empirically, SGD will easily get stuck at bad local minima as well [109].

Inspired by the structure of residual network (ResNet) [51], we add an extra identity mapping for the hidden layer (see Figure 5.1). Surprisingly, we show that simply by adding this mapping, with the standard initialization scheme and small step size, SGD always converges to the ground truth. In other words, the optimization becomes significantly easier, after adding the identity mapping. See Figure 5.2, based on our analysis, the region near the identity matrix $\mathbf{I}$ contains only one global minimum without any saddle points or local minima, thus is easy for SGD to optimize. The role of the identity mapping here, is to move the initial point to this easier region (better initialization).

Other than being feedforward and shallow, our network is different from ResNet in the sense that our identity mapping skips one layer instead of two. However, as we will show in Section 5.5.1, the skip-one-layer identity mapping already brings significant improvement to vanilla networks.

Formally, we consider the following function.

$$f(x, \mathbf{W}) = \|\mathrm{ReLU}((\mathbf{I} + \mathbf{W})^\top x)\|_1 \tag{5.1}$$

where $\mathrm{ReLU}(v) = \max(v, 0)$ is the ReLU activation function. $x \in \mathbb{R}^d$ is the input vector sampled

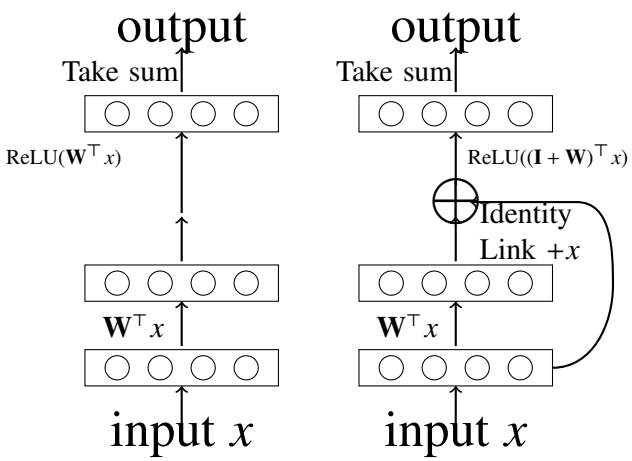


Figure 5.1: Vanilla network (left), with identity mapping (right)

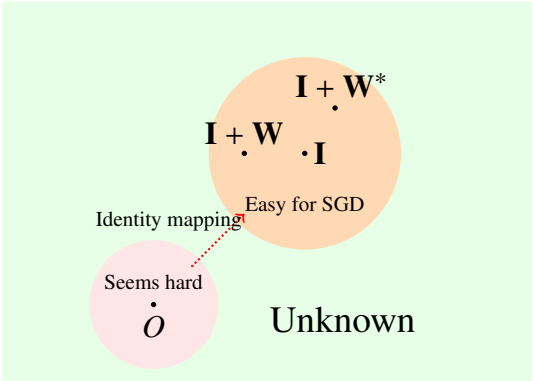


Figure 5.2: Illustration for our result.

from a Gaussian distribution, and $\mathbf{W} \in \mathbb{R}^{d\times d}$ is the weight matrix, where d is the number of input units. Notice that $\mathbf{I}$ adds e_i to column i of $\mathbf{W}$, which makes f *asymmetric* in the sense that by switching any two columns in $\mathbf{W}$, we get different functions.

Following the standard setting [108, 124], we assume that there exists a two-layer teacher network with weight $\mathbf{W}^*$. We train the student network using ℓ_2 loss:

$$\mathsf{L}(\mathbf{W}) = \mathbb{E}_x[(f(x,\mathbf{W}) - f(x,\mathbf{W}^*))^2] \tag{5.2}$$

We will define a potential function g, and show that if g is small, the gradient points to partially correct direction and we get closer to $\mathbf{W}^*$ after every SGD step. However, g could be large and thus

gradient might point to the reverse direction. Fortunately, we also show that if g is large, by doing SGD, it will keep decreasing until it is small enough while maintaining the weight $\mathbf{W}$ in a nice region. We call the process of decreasing g as Phase I, and the process of approaching $\mathbf{W}^*$ as Phase II. See Figure 5.3 and simulations in Section 5.5.3.

Our two phases framework is fundamentally different from any type of local convergence, as in Phase I, the gradient is pointing to the wrong direction to $\mathbf{W}^*$, so the path from $\mathbf{W}$ to $\mathbf{W}^*$ is non-convex, and SGD takes a long detour to arrive $\mathbf{W}^*$. This framework could be potentially useful for analyzing other non-convex problems.

To support our theory, we have done a few other experiments and got interesting observations. For example, as predicted by our theorem, we found that for multilayer feedforward network with identity mappings, zero initialization performs as good as random initialization. At the first glance, it contradicts the common belief "random initialization is necessary to break symmetry", but actually the identity mapping itself serves as the asymmetric component. See Section 5.5.4.

Another common belief is that neural network has lots of local minima and saddle points [27], so even if there exists a global minimum, we may not be able to arrive there. As a result, even when the teacher network is shallow, the student network usually needs to be deeper, otherwise it will underfit. However, both our theorem and our experiment show that if the shallow teacher network is in a pretty large region near identity (Figure 5.2), SGD always converges to the global minimum by initializing the weights $\mathbf{I} + \mathbf{W}$ in this region, with equally shallow student network. By contrast, wrong initialization gets stuck at local minimum and underfit. See Section 5.5.2.

Related Work

Expressivity. Even two-layer network has great expressive power. For example, two-layer network with sigmoid activations could approximate any continuous function [55, 26, 9]. ReLU is the state-of-the-art activation function [93, 38], and has great expressive power as well [91, 99, 98, 17, 75].

Learning. Most previous results on learning neural network are negative [115, 84, 114], or positive but with algorithms other than SGD [66, 131, 113, 39, 40, 41], or with strong assumptions on the model [4, 5]. [110] proved that with high probability, there exists a continuous decreasing path from random initial point to the global minimum, but SGD may not follow this path. Recently, Zhong et al. showed that with initialization point found using tensor decomposition, gradient descent could find the ground truth for one hidden layer network [133].

Linear network and independent activation. Some previous works simplified the model by ignoring the activation functions and considering deep linear networks [111, 70] or deep linear residual networks [45], which can only learn linear functions. Some results are based on independent activation assumption that the activations of ReLU are independent of each other [18], or are independent to the input [22, 70].

Saddle points. It is observed that saddle point is not a big problem for neural networks [27, 42]. In general, if the objective is strict-saddle [35], SGD could escape all saddle points.

5.2 Preliminaries

Denote x as the input vector in $\mathbb{R}^d$. For now, we first consider x sampled from normal distribution $\mathcal{N}(0, \mathbf{I})$. Denote $\mathbf{W}^* = (w_1^*, \cdots, w_n^*) \in \mathbb{R}^{d\times d}$ as the weights for the teacher network,

$\mathbf{W} = (w_1, \cdots, w_n) \in \mathbb{R}^{d\times d}$ as the weights for the student network, where $w_i^*, w_i \in \mathbb{R}^d$ are column vectors. $f(x, \mathbf{W}^*)$, $f(x, \mathbf{W})$ are defined in (5.1), representing the teacher and student network.

We want to know whether a randomly initialized $\mathbf{W}$ will converge to $\mathbf{W}^*$, if we run SGD with l_2 loss defined in (5.2). Alternatively, we can write the loss $\mathsf{L}(\mathbf{W})$ as

$$\mathbb{E}_x[(\Sigma_i \mathrm{ReLU}(\langle e_i + w_i, x\rangle) - \Sigma_i \mathrm{ReLU}(\langle e_i + w_i^*, x\rangle))^2]$$

Taking derivative with respect to w_j, we get

$$\nabla \mathsf{L}(\mathbf{W})_j = 2\mathbb{E}_x\left[\left(\sum_i \mathrm{ReLU}(\langle e_i + w_i, x\rangle) - \sum_i \mathrm{ReLU}(\langle e_i + w_i^*, x\rangle)\right) x \mathbb{1}_{\langle e_j + w_j, x\rangle \geq 0}\right]$$

where $\mathbb{1}_e$ is the indicator function that equals 1 if the event e is true, and 0 otherwise. Here $\nabla \mathsf{L}(\mathbf{W}) \in \mathbb{R}^{d\times d}$, and $\nabla \mathsf{L}(\mathbf{W})_j$ is its j-th column.

Denote $\theta_{i,j}$ as the angle between $e_i + w_i$ and $e_j + w_j$, $\theta_{i^*,j}$ as the angle between $e_i + w_i^*$ and $e_j + w_j$. Denote $\bar{v} = \frac{v}{\|v\|_2}$. Denote $\overline{\mathbf{I} + \mathbf{W}^*}$ and $\overline{\mathbf{I} + \mathbf{W}^*}$ as the column-normalized version of $\mathbf{I} + \mathbf{W}^*$ and $\mathbf{I} + \mathbf{W}$ such that every column has unit norm. Since the input is from a normal distribution, one can compute the expectation inside the gradient as follows.

Lemma 5.2.1 (Eqn (13) from [124]). *If* $x \sim \mathcal{N}(0, \mathbf{I})$, *then* $-\nabla \mathsf{L}(\mathbf{W})_j = \sum_{i=1}^d \left(\frac{\pi}{2}(w_i^* - w_i) + \left(\frac{\pi}{2} - \theta_{i^*,j}\right)(e_i + w_i^*) - \left(\frac{\pi}{2} - \theta_{i,j}\right)(e_i + w_i) + (\|e_i + w_i^*\|_2 \sin\theta_{i^*,j} - \|e_i + w_i\|_2 \sin\theta_{i,j})\overline{e_j + w_j}\right)$

Remark. Although the gradient of ReLU is not well defined at the point of zero, if we assume input x is from the Gaussian distribution, the loss function becomes smooth, and the gradient is well defined everywhere.

Denote $u \in \mathbb{R}^d$ as the all one vector. Denote $\mathrm{Diag}(\mathbf{W})$ as the diagonal matrix of matrix $\mathbf{W}$, $\mathrm{Diag}(v)$ as a diagonal matrix whose main diagonal equals to the vector v. Denote $\text{Off-Diag}(\mathbf{W}) \triangleq \mathbf{W} - \mathrm{Diag}(\mathbf{W})$. Denote $[d]$ as the set $\{1, \cdots, d\}$. Throughout the paper, we abuse the notation of

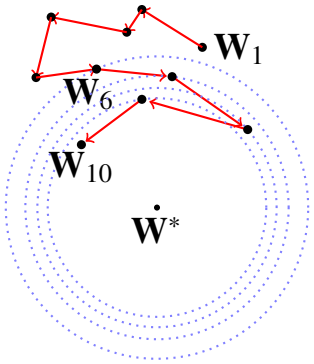


Figure 5.3: Phase I: $\mathbf{W}_1 \to \mathbf{W}_6$, $\mathbf{W}$ may go to the wrong direction but the potential is shrinking. Phase II: $\mathbf{W}_6 \to \mathbf{W}_{10}$, $\mathbf{W}$ gets closer to $\mathbf{W}^*$ in every step by one point convexity.

inner product between matrices $\mathbf{W}, \mathbf{W}^*, \nabla \mathsf{L}(\mathbf{W})$, such that $\langle \nabla \mathsf{L}(\mathbf{W}), \mathbf{W} \rangle$ means the summation of the entrywise products. $\|\mathbf{W}\|_2$ is the spectral norm of $\mathbf{W}$, and $\|\mathbf{W}\|_F$ is the Frobenius norm of $\mathbf{W}$. We define the potential function g and variables $g_j, \mathbf{A}_j, \mathbf{A}$ below, which will be useful in the proof.

Definition 5.2.2. *We define the potential function* $g \triangleq \sum_{i=1}^{d} (\|e_i + w_i^*\|_2 - \|e_i + w_i\|_2)$*, and variable* $g_j \triangleq \sum_{i \neq j} (\|e_i + w_i^*\|_2 - \|e_i + w_i\|_2)$.

Definition 5.2.3. *Denote* $\mathbf{A}_j \triangleq \sum_{i \neq j} ((e_i + w_i^*)\overline{e_i + w_i^*}^\top - (e_i + w_i)\overline{e_i + w_i}^\top)$, $\mathbf{A} \triangleq \sum_{i=1}^{d} ((e_i + w_i^*)\overline{e_i + w_i^*}^\top - (e_i + w_i)\overline{e_i + w_i}^\top) = (\mathbf{I} + \mathbf{W}^*)\overline{\mathbf{I} + \mathbf{W}^*}^\top - (\mathbf{I} + \mathbf{W})\overline{\mathbf{I} + \mathbf{W}}^\top$.

Assume $\mathbf{W}_0$ is the initial point, and in step $t > 0$, we have the following SGD updating rule:

$$\mathbf{W}_{t+1} = \mathbf{W}_t - \eta_t \mathbf{G}_t$$

where the stochastic gradient $\mathbf{G}_t = \nabla \mathsf{L}(\mathbf{W}_t) + \mathbf{E}_t$ with $\mathbb{E}[\mathbf{E}_t] = \mathbf{0}$ and $\|\mathbf{E}_t\|_F \leq \varepsilon$. Let $G_2 \triangleq 6d\gamma + \varepsilon, G_F \triangleq 6d^{1.5}\gamma + \varepsilon$, where γ is the upper bound of $\|\mathbf{W}^*\|_2$ and $\|\mathbf{W}_0\|_2$ (defined later). As we will see in Lemma B.3.2, they are the upper bound of $\|\mathbf{G}_t\|_2$ and $\|\mathbf{G}_t\|_F$ respectively.

5.3 Main theorem

Theorem 5.3.1 (Main Theorem)**.** *There exists* constants $\gamma > \gamma_0 > 0$ *such that If* $x \sim \mathcal{N}(0, \mathbf{I})$, $\|\mathbf{W}_0\|_2, \|\mathbf{W}^*\|_2 \leq \gamma_0$, $d \geq 100$, $\varepsilon \leq \gamma^2$, *then SGD for* $\mathsf{L}(\mathbf{W})$ *will find the ground truth* $\mathbf{W}^*$ *by two phases. In Phase I, by setting* $\eta \leq \frac{\gamma^2}{G_2^2}$, *the potential function will keep decreasing until it is smaller than* $197\gamma^2$, *which takes at most* $\frac{1}{16\eta}$ *steps. In Phase II, for any* $\alpha > 0$ *and any* T *such that* $T^\alpha \log T \geq \frac{36d}{100^4(1+\alpha)G_F^2}$, *if we set* $\eta = \frac{(1+\alpha)\log T}{\delta T}$, *we have* $\mathbb{E}\|\mathbf{W}_T - \mathbf{W}^*\|_F^2 \leq \frac{20000(1+\alpha)\log T G_F^2}{9T}$.

Remarks. Randomly initializing the weights with $O(1/\sqrt{d})$ is standard in deep learning, see [78, 37, 50]. It is also well known that if the entries are initialized with $O(1/\sqrt{d})$, the spectral norm of the random matrix is $O(1)$ [106]. So our result matches with the common practice. Moreover, as we will show in Section 5.5.5, networks with small average spectral norm already have good performance. Thus, our assumption $\|\mathbf{W}^*\|_2 = O(1)$ is reasonable. Notice that here we assume the *spectral norm* of $\mathbf{W}^*$ to be constant, which means the Frobenius norm $\|\mathbf{W}^*\|_F$ could be as big as $O(\sqrt{d})$.

The assumption that the input follows a Gaussian distribution is not necessarily true in practice (Although this is a common assumption appeared in the previous papers [22, 124, 129], and also considered plausible in [23]). We could easily generalize the analysis to rotation invariant distributions, and potentially more general distributions (see Section 5.6). Moreover, previous analyses either ignore the nonlinear activations and thus consider linear model [111, 70, 45], or directly [22, 70] or indirectly [124][1] assume that the activations are independent. By contrast, in our model the ReLU activations are highly correlated[2] as $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 = \Omega(1)$. As pointed out by [23], eliminating the unrealistic assumptions on activation independence is the central problem

[1] They assume input is Gaussian and the $\mathbf{W}^*$ is orthonormal, which means the activations are independent in teacher network.

[2] Let σ_i be the output of i-th ReLU unit, then in our setting, $\sum_{i,j} \mathrm{Cov}[\sigma_i, \sigma_j]$ can be as large as $\Omega(d)$, which is far from being independent.

of analyzing the loss surface of neural network, which was not fully addressed by the previous analyses.

To prove the main theorem, we split the process and present the following two theorems, which will be proved in Appendix B.3 and B.4.

Theorem 5.3.2 (Phase I)**.** *There exists a constant $\gamma > \gamma_0 > 0$ such that If $\|\mathbf{W}_0\|_2, \|\mathbf{W}^*\|_2 \leq \gamma_0$, $d \geq 100$, $\eta \leq \frac{\gamma^2}{G_2^2}$, $\varepsilon \leq \gamma^2$, then g_t will keep decreasing by a factor of $1 - 0.5\eta d$ for every step, until $g_{t_1} \leq 197\gamma^2$ for step $t_1 \leq \frac{1}{16\eta}$. After that, Phase II starts. That is, for every $T > t_1$, we have $\|\mathbf{W}_T\|_2 \leq \frac{1}{100}$ and $g_T \leq 0.1$.*

Theorem 5.3.3 (Phase II)**.** *There exists a constant γ such that if $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \leq \gamma$, and $g \leq 0.1$, then $\langle -\nabla \mathsf{L}(\mathbf{W}), \mathbf{W}^* - \mathbf{W} \rangle = \sum_{j=1}^{d} \langle -\nabla \mathsf{L}(\mathbf{W})_j, w_j^* - w_j \rangle > 0.03\|\mathbf{W}^* - \mathbf{W}\|_F^2$.*

With these two theorems, we get the main theorem immediately.

Proof for Theorem 5.3.1. By Theorem 5.3.2, we know the statement for Phase I is true, and we will enter phase II in $\frac{1}{16\eta}$ steps. After entering Phase II, based on Theorem 5.3.3, we simply use Lemma 2.3.6 by setting $\delta = 0.03$, $D = \frac{\sqrt{d}}{50}, G = G_F$ to get the convergence guarantee. □

5.4 Overview of the proofs

General Picture. In many convergence analyses for non-convex functions, one would like to show that L is one point strongly convex, and directly apply Lemma 2.3.6 to get the convergence result. However, this is not true for 2-layer neural network, as the gradient may point to the wrong direction, see Section 5.5.3.

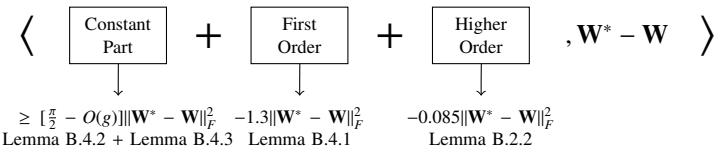


Figure 5.4: Lower bounds of inner product using Taylor expansion

So when is our L one point convex? Consider the following thought experiment: First, suppose $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \to 0$, we know $\|w_i\|_2, \|w_i^*\|_2$ also go to 0. Thus, $e_i + w_i$ and $e_i + w_i^*$ are close to e_i. As a result, $\theta_{i,j}, \theta_{i^*,j} \approx \frac{\pi}{2}$, and $\theta_{i^*,i} \approx 0$. Based on Lemma 5.2.1, this gives us a naïve approximation of the negative gradient, i.e., $-\nabla\mathsf{L}(\mathbf{W})_j \approx \frac{\pi}{2}(w_j^* - w_j) + \frac{\pi}{2}\sum_{i=1}^d (w_i^* - w_i) + \overline{e_j + w_j}\sum_{i\neq j}(\|e_i + w_i^*\|_2 - \|e_i + w_i\|_2)$
.

While the first two terms $\frac{\pi}{2}(w_j^* - w_j)$ and $\frac{\pi}{2}\sum_{i=1}^d (w_i^* - w_i)$ have positive inner product with $\mathbf{W}^* - \mathbf{W}$, the last term $g_j = \overline{e_j + w_j}\sum_{i\neq j}(\|e_i + w_i^*\|_2 - \|e_i + w_i\|_2)$ can point to arbitrary direction. If the last term is small, it can be covered by the first two terms, and L becomes one point strongly convex. So we define a potential function closely related to the last term: $g = \sum_{i=1}^d (\|e_i + w_i^*\|_2 - \|e_i + w_i\|_2)$. We show that if g is small enough, L is also one point strongly convex (Theorem 5.3.3).

However, from random initialization, g can be as large as of $\Omega(\sqrt{d})$, which is too big to be covered. Fortunately, we show that if g is big, it will gradually decrease simply by doing SGD on L. More specifically, we introduce a *two phases* convergence analysis framework:

1. In Phase I, the potential function g is decreasing to a small value.
2. In Phase II, g remains small, so L is one point convex and thus $\mathbf{W}$ starts to converge to $\mathbf{W}^*$.

We believe that this framework could be helpful for other non-convex problems.

Technical difficulty: Phase I. Our key technical challenge is to show that in Phase I, the potential function actually decreases to $O(1)$ after polynomial number of iterations. However, we cannot show this by merely looking at g itself. Instead, we introduce an auxiliary variable

$s = (\mathbf{W}^* - \mathbf{W})u$, where u is the all one vector. By doing a careful calculation, we get their joint update rules (Lemma B.3.3 and Lemma B.3.4):

$$\begin{cases} s_{t+1} & \approx s_t - \frac{\pi\eta d}{2} s_t + \eta O(\sqrt{d} g_t + \sqrt{d}\gamma) \\ g_{t+1} & \approx g_t - \eta d g_t + \eta O(\gamma\sqrt{d}\|s_t\|_2 + d\gamma^2) \end{cases}$$

Solving this dynamics, we can show that g_t will approach to (and stay around) $O(\gamma)$, thus we enter Phase II.

Technical difficulty: Phase II. Although the overall approximation in the thought experiment looks simple, the argument is based on an over simplified assumption that $\theta_{i^*,j}, \theta_{i,j} \approx \frac{\pi}{2}$ for $i \neq j$. However, when $\mathbf{W}^*$ has constant spectral norm, even when $\mathbf{W}$ is very close to $\mathbf{W}^*$, θ_{i,j^*} could be constantly far away from $\frac{\pi}{2}$, which prevents us from applying this approximation directly. To get a formal proof, we use the standard Taylor expansion and control the higher order terms. Specifically, we write $\theta_{i^*,j}$ as $\theta_{i^*,j} = \arccos\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle$ and expand arccos at point 0, thus,

$$\theta_{i^*,j} = \frac{\pi}{2} - \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle + O(\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^3)$$

However, even when $\mathbf{W} \approx \mathbf{W}^*$, the higher order term $O(\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^3)$ still can be as large as a constant, which is too big for us. Our trick here is to consider the "joint Taylor expansion":

$$\theta_{i^*,j} - \theta_{i,j} = \langle \overline{e_i + w_i} - \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle + O(|\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^3 - \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^3|)$$

As $\mathbf{W}$ approaches $\mathbf{W}^*$, $|\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^3 - \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^3|$ also tends to zero, therefore our approximation has bounded error.

In the thought experiment, we already know that the constant part in the Taylor expansion of $\nabla \mathsf{L}(\mathbf{W})$ is $\frac{\pi}{2} - O(g)$-one point convex. We show that after taking inner product with $\mathbf{W}^* - \mathbf{W}$, the first order terms are lower bounded by (roughly) $-1.3\|\mathbf{W}^* - \mathbf{W}\|_F^2$ and the higher order terms are lower bounded by $-0.085\|\mathbf{W}^* - \mathbf{W}\|_F^2$. Adding them together, we can see that $\mathsf{L}(\mathbf{W})$ is one point convex as long as g is small. See Figure 5.4.

Table 5.1: Test error of three 56-layer networks on Cifar-10

	ResNet	Single skip	Vanilla
Test Err	6.97%	9.01%	12.04%

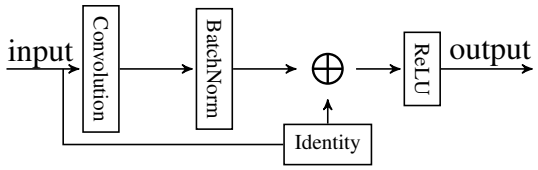


Figure 5.5: Illustration of one block in single skip model in Sec 5.5.1

Geometric Lemma. In order to get through the whole analysis, we need tight bounds on a few common terms that appear everywhere. Instead of using naïve algebraic techniques, we come up with a nice geometric proof to get nearly optimal bounds. See Appendix B.5.

5.5 Experiments

In this section, we present several simulation results to support our theory.

5.5.1 Importance of identity mapping

In this experiment, we compare the standard ResNet [51] and *single skip model* where identity mapping skips only one layer. See Figure 5.5 for the single skip model. We also ran the vanilla network, where the identity mappings are completely removed.

In this experiment, we choose Cifar-10 as the dataset, and all the networks have 56-layers. Other than the identity mappings, all other settings are identical and default. We run the experiments for 5 times and report the average test error. As we can see in Table 5.1, compared with vanilla network, by simply using a single skip identity mapping, one can already improve the test error by 3.03%, and is 2.04% close to the ResNet. So single skip identity mapping brings significant improvement on test accuracy.

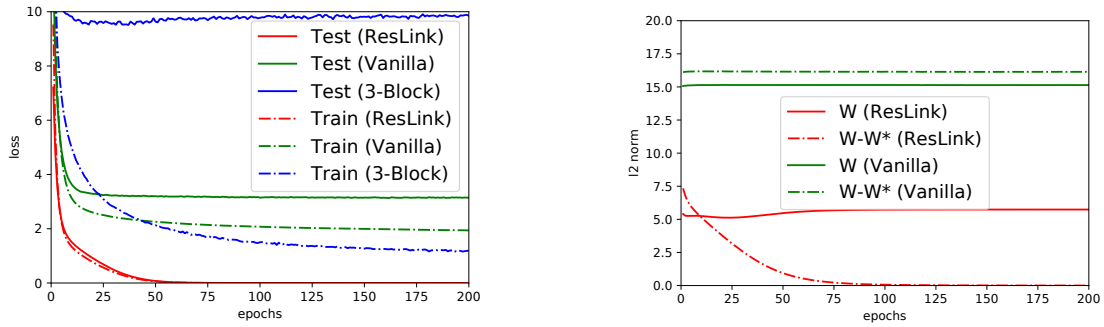
5.5.2 Global minimum convergence

In this experiment, we verify our main theorem that for two-layer teacher network and student network with identity mappings, as long as $\|\mathbf{W}_0\|_2, \|\mathbf{W}^*\|_2$ is small, SGD always converges to the global minimum $\mathbf{W}^*$, thus gives almost 0 training error and test error. We consider three student networks. The first one (ResLink) is defined using (5.2), the second one (Vanilla) is the same model without the identity mapping. The last one (3-Block) is a three block network with each block containing a linear layer (500 hidden nodes), a batch normalization and a ReLU layer. The teacher network always shares the same structure as the student network.

The input dimension is 100. We generated a fixed $\mathbf{W}^*$ for all the trials with $\|\mathbf{W}^*\|_2 \approx 0.6, \|\mathbf{W}^*\|_F \approx 5.7$. We generated a training set of size $100,000$, and test set of size $10,000$, sampled from a Gaussian distribution. We use batch size 200, step size 0.001. We run ResLink for 5 times with random initialization ($\|\mathbf{W}\|_2 \approx 0.6$ and $\|\mathbf{W}\|_F \approx 5$), and plot the curves by taking the average.

Figure 5.6a shows test error and training error of the three networks. Comparing Vanilla with 3-Block, we find that 3-Block is more expressive, so its training error is smaller compared with vanilla network; but it suffers from overfitting and has bigger test error. This is the standard overfitting vs underfitting tradeoff. Surprisingly, with only one hidden layer, ResLink has both zero test error and training error. If we look at Figure 5.6b, we know the distance between $\mathbf{W}$ and $\mathbf{W}^*$ converges to 0, meaning ResLink indeed finds the global optimal in all 5 trials. By contrast, for vanilla network, which is essentially the same network with different initialization, $\|\mathbf{W} - \mathbf{W}^*\|_2$ does not converge to zero[3]. This is exactly what our theory predicted.

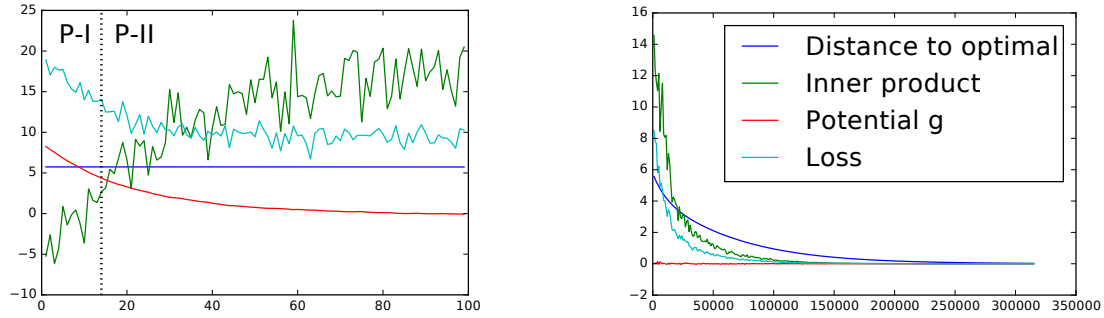
[3]To make comparison meaningful, we set $\mathbf{W} - \mathbf{I}$ to be the actual weight for Vanilla as its identity mapping is missing, which is why it has a much bigger initial norm.



(a) Test Error, Train Error

(b) $\|\mathbf{W}^* - \mathbf{W}\|_F, \|\mathbf{W}\|_F$

Figure 5.6: Verifying the global convergence



(a) First 100 iterations

(b) The entire process

Figure 5.7: Verifying the dynamics

5.5.3 Verify the dynamics

In this experiment, we verify our claims on the dynamics. Based on the analysis, we construct a 1500×1500 matrix $\mathbf{W}$ s.t. $\|\mathbf{W}\|_2 \approx 0.15, \|\mathbf{W}\|_F \approx 5$, and set $\mathbf{W}^* = 0$. By plugging them into (5.2), one can see that even in this simple case that $\mathbf{W}^* = 0$, initially the gradient is pointing to the wrong direction, i.e., not one point convex. We then run SGD on $\mathbf{W}$ by using samples x from Gaussian distribution, with batch size 300, step size 0.0001.

Figure 5.7a shows the first 100 iterations. We can see that initially the inner product defined in Definition 2.3.5 is negative, then after about 15 iterations, it turns positive, which means $\mathbf{W}$ is in the one point strongly convex region. At the same time, the potential g keeps decreasing to a small

value, while the distance to optimal (which also equals to $\|\mathbf{W}\|_F$ in this experiment) is not affected. They precisely match with our description of Phase I in Theorem 5.3.2.

After that, we enter Phase II and slowly approach to $\mathbf{W}^*$, see Figure 5.7b. Notice that the potential g is always very small, the inner product is always positive, and the distance to optimal is slowly decreasing. Again, they precisely match with our Theorem 5.3.3.

5.5.4 Zero initialization works

In this experiment, we used a simple 5-block neural network on MNIST, where every block contains a $784 * 784$ feedforward layer, an identity mapping, and a ReLU layer. Cross entropy criterion is used. We compare zero initialization with standard $O(1/\sqrt{d})$ random initialization. We found that for zero initialization, we can get 1.28% test error, while for random initialization, we can get 1.27% test error. Both results were obtained by taking average among 5 runs and use step size 0.1, batch size 256. If the identity mapping is removed, zero initialization no longer works.

5.5.5 Spectral norm of $\mathbf{W}^*$

We also applied the exact model f defined in (5.1) to distinguish two classes in MNIST. For any input image x, We say itâĂŹs in class A if $f(x, \mathbf{W}) < T_{A,B}$, and in class B otherwise. Here $T_{A,B}$ is the optimal threshold for the function $f(x, \mathbf{0})$ to distinguish A and B. If $\mathbf{W} = \mathbf{0}$, we get 7% training error for distinguish class 0 and class 1. However, it can be improved to 1% with $\|\mathbf{W}\|_2 = 0.6$. We tried this experiment for all possible 45 pairs of classes in MNIST, and improve the average training error from 34% (using $\mathbf{W} = \mathbf{0}$) to 14% (using $\|\mathbf{W}\|_2 = 0.6$). Therefore our model with $\|\mathbf{W}\|_2 = \Omega(1)$ has reasonable expressive power, and is substantially different from just using the identity mapping

alone.

5.6 Discussions

The assumption that the input is Gaussian can be relaxed in several ways. For example, when the distribution is $\mathcal{N}(0, \Sigma)$ where $\|\Sigma - \mathbf{I}\|_2$ is bounded by a small constant, the same result holds with slightly worse constants. Moreover, since the analysis relies Lemma 5.2.1, which is proved by converting the original input space into polar space, it is easy to generalize the calculation to rotation invariant distributions. Finally, for more general distributions, as long as we could explicitly compute the expectation, which is in the form of $O(\mathbf{W}^* - \mathbf{W})$ plus certain potential function, our analysis framework may also be applied.

CHAPTER 6

HYPERPARAMETER TUNING: HARMONICA

6.1 Introduction

Large scale machine learning and optimization systems usually involve a large number of free parameters for the user to fix according to their application. A timely example is the training of deep neural networks for a signal processing application: the ML specialist needs to decide on an architecture, depth of the network, choice of connectivity per layer (convolutional, fully-connected, etc.), choice of optimization algorithm and recursively choice of parameters inside the optimization library itself (learning rate, momentum, etc.).

Given a set of hyperparameters and their potential assignments, the naive practice is to search through the entire grid of parameter assignments and pick the one that performed the best, a.k.a. “grid search". As the number of hyperparameters increases, the number of possible assignments increases exponentially and a grid search becomes quickly infeasible. It is thus crucial to find a method for automatic tuning of these parameters.

This auto-tuning, or finding a good setting of these parameters, is now referred to as hyperparameter optimization (HPO), or simply automatic machine learning (auto-ML). For continuous hyperparameters, gradient descent is usually the method of choice [87, 86, 33]. Discrete parameters, however, such as choice of architecture, number of layers, connectivity and so forth are significantly more challenging. More formally, let

$$f : \{-1, 1\}^n \mapsto [0, 1]$$

be a function mapping hyperparameter choices to test error of our model. That is, each dimension corresponds to a certain hyperparameter (number of layers, connectivity, etc.), and for simplicity of

illustration we encode the choices for each parameter as binary numbers $\{-1, 1\}$. The goal of HPO is to approximate the minimizer $x^* = \text{argmin}_{x\in\{0,1\}^n} f(x)$ in the following setting:

1. Oracle model: evaluation of f for a given choice of hyperparameters is assumed to be very expensive. Such is the case of training a given architecture of a huge dataset.
2. Parallelism is crucial: testing several model hyperparameters in parallel is entirely possible in cloud architecture, and dramatically reduces overall optimization time.
3. f is structured.

The third point is very important since clearly HPO is information-theoretically hard and 2^n evaluations of the function are necessary in the worst case. Different works have considered exploiting one or more of the properties above. The approach of Bayesian optimization [117] addresses the structure of f, and assumes that a useful prior distribution over the structure of f is known in advance. Multi-armed bandit algorithms [81], and Random Search [12], exploit computational parallelism very well, but do not exploit any particular structure of f. These approaches are surveyed in more detail later.

6.1.1 Our contribution

In this chapter we introduce a new *spectral* approach to hyperparameter optimization. Our main idea is to make assumptions on the structure of f in the Fourier domain. Specifically we assume that f can be approximated by a sparse and low degree polynomial in the Fourier basis. This means intuitively that it can be approximated well by a decision tree.

The implication of this assumption is that we can obtain a rigorous theoretical guarantee: approximate minimization of f over the boolean hypercube with **function evaluations only linear**

in sparsity that can be carried out in parallel. We further give improved heuristics on this basic construction and show experiments showing our assumptions are validated in practice for HPO as applied to deep learning over image datasets.

Thus our contributions can be listed as:

- A new spectral method called *Harmonica* that has provable guarantees: sample-efficient recovery if the underlying hyperparameter objective is a sparse (noisy) polynomial and easy to implement on parallel architectures.
- We demonstrate significant improvements in accuracy, sample complexity, and running time for deep neural net training experiments. We compare ourselves to state-of-the-art solvers from Bayesian optimization, Multi-armed bandit techniques, and Random Search. Projecting to even higher numbers of hyperparameters, we perform simulations that show several orders-of-magnitude of speedup versus Bayesian optimization techniques.
- Improved bounds on the sample complexity of learning noisy, size s decision trees over n variables under the uniform distribution. We observe that the classical sample complexity bound of $n^{O(\log(s/\varepsilon))}$ due to [83] can be improved to quadratic in the size of the tree $\tilde{O}(s^2/\varepsilon \cdot \log n)$ while matching the best known quasipolynomial bound in running time.

6.1.2 Related work

The literature on discrete-domain HPO can be roughly divided into two: probabilistic approaches and decision-theoretic methods. In critical applications, researchers usually use a grid search over all parameter space, but that becomes quickly prohibitive as the number of hyperparameter grows. Gradient-based methods such as [87, 86, 33, 10] are applicable only to continuous hyperparameters

which we do not consider. Neural network structural search based on reinforcement learning is an active direction [8, 135, 134], which usually needs many samples of network architectures.

Probabilistic methods and Bayesian optimization. Bayesian optimization (BO) algorithms [13, 117, 123, 118, 34, 126, 61] tune hyperparameters by assuming a prior distribution of the loss function, and then keep updating this prior distribution based on the new observations. Each new observation is selected according to an acquisition function, which balances exploration and exploitation such that the new observation gives us a better result, or helps gain more information. The BO approach is inherently serial and difficult to parallelize, and its theoretical guarantees have thus far been limited to statistical consistency (convergence in the limit).

Decision-theoretic methods. Perhaps the simplest approach to HPO is random sampling of different choices of parameters and picking the best amongst the chosen evaluations [12]. It is naturally very easy to implement and parallelize. Upon this simple technique, researchers have tried to allocate different budgets to the different evaluations, depending on their early performance. Using adaptive resource allocation techniques found in the multi-armed bandit literature, Successive Halving (SH) algorithm was introduced [65]. Hyperband further improves SH by automatically tuning the hyperparameters in SH [81].

Learning decision trees. Prior work for learning decision trees (more generally Boolean functions that are approximated by low-degree polynomials) used the celebrated "low-degree" algorithm of [83]. Their algorithm uses random sampling to estimate each low-degree Fourier coefficient to high accuracy.

We make use of the approach of [120], who showed how to learn low-degree, sparse Boolean functions using tools from compressed sensing (similar approaches were taken by [76] and [94]).

We observe that their approach can be extended to learn functions that are both "approximately sparse" (in the sense that the L_1 norm of the coefficients is bounded) and "approximately low-degree" (in the sense that most of the L_2 mass of the Fourier spectrum resides on monomials of low-degree). This implies the first decision tree learning algorithm with polynomial sample complexity that handles adversarial noise. In addition, we obtain the optimal dependence on the error parameter ε.

For the problem of learning *exactly* k-sparse Boolean functions over n variables, [47] have recently shown that $O(nk \log n)$ uniformly random samples suffice. Their result is not algorithmic but does provide an upper bound on the information-theoretic problem of how many samples are required to learn. The best algorithm in terms of running time for learning k-sparse Boolean functions is due to [31], and requires time $2^{\Omega(n/\log n)}$. It is based on the [15] algorithm for learning parities with noise.

Techniques. Our methods are heavily based on known results from the analysis of boolean functions as well as compressed sensing.

6.2 Setup and definitions

The problem of hyperparameter optimization is that of minimizing a discrete, real-valued function, which we denote by $f : \{-1, 1\}^n \mapsto [-1, 1]$ (we can handle arbitrary inputs, binary is chosen for simplicity of presentation).

In the context of hyperparameter optimization, function evaluation is very expensive, although parallelizable, as it corresponds to training a deep neural net. In contrast, any computation that does not involve function evaluation is considered less expensive, such as computations that require time $\Omega(n^d)$ for "somewhat large" d or are subexponential (we still consider runtimes that are exponential

in n to be costly).

6.2.1 Basics of Fourier analysis

The reader is referred to [96] for an in depth treatment of Fourier analysis of Boolean functions. Let $f : \mathbf{X} \mapsto [-1, 1]$ be a function over domain $\mathbf{X} \subseteq \mathbb{R}^n$. Let $\mathcal{D}$ a probability distribution on $\mathcal{X}$. We write $g \equiv_\varepsilon f$ and say that f, g are **E-close** if

$$\mathbb{E}_{x\sim\mathbb{D}}[(f(x) - g(x))^2] \leq \mathbf{E}.$$

Definition 6.2.1. *[103] We say a family of functions $\psi_1, \ldots, \psi_N$ (ψ_i maps $\mathcal{X}$ to $\mathbb{R}$) is a* Random Orthonormal Family *with respect to $\mathcal{D}$ if*

$$\mathbb{E}_{\mathbb{D}}[\psi_i(X) \cdot \psi_j(X)] = \delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}.$$

The expectation is taken with respect to probability distribution $\mathcal{D}$. We say that the family is K-bounded if $\sup_{x\in\mathcal{X}} |\psi_i(x)| \leq K$ for every i. Henceforth we assume $K = 1$.

An important example of a random orthonormal family is the class of parity functions with respect to the uniform distribution on $\{-1, 1\}^n$:

Definition 6.2.2. *A parity function on some subset of variables $S \subseteq [n]$ is the function $\chi_S : \{-1, 1\}^n \mapsto \{-1, 1\}$ where $\chi_S(x) = \prod_{i\in S} x_i$.*

It is easy to see that the set of all 2^n parity functions $\{\chi_S\}$, one for each $S \subseteq [n]$, form a random orthonormal family with respect to the uniform distribution on $\{-1, 1\}^n$.

This random orthonormal family is often referred to as the Fourier basis, as it is a complete orthonormal basis for the class of Boolean functions with respect to the uniform distribution on

$\{-1,1\}^n$. More generally, for any $f : \{-1,1\}^n \mapsto \mathbb{R}$, f can be uniquely represented in this basis as

$$f(x) = \sum_{S \subseteq [n]} \hat{f}_S \chi_S(x)$$

where

$$\hat{f}_S = \langle f, \chi_S \rangle = \mathbb{E}_{x \in \{-1,1\}^n}[f(x)\chi_S(x)]$$

is the Fourier coefficient corresponding to S where x is drawn uniformly from $\{-1,1\}^n$. We also have Parseval's identity: $\mathbb{E}[f^2] = \sum_S \hat{f}_S^2$.

In this paper, we will work exclusively with the above parity basis. Our results apply more generally, however, to any orthogonal family of polynomials (and corresponding product measure on $\mathbb{R}^n$). For example, if we wished to work with continuous hyperparameters, we could work with families of Hermite orthogonal polynomials with respect to multivariate spherical *Gaussian* distributions.

We conclude with a definition of low-degree, approximately sparse (bounded L_1 norm) functions:

Definition 6.2.3 (Approximately sparse function)**.** *Let $\{\chi_S\}$ be the parity basis, and let C be a class of functions mapping $\{-1,1\}$ to $\mathbb{R}$. Thus for $f \in C$, $f = \sum_S \hat{f}(S)\chi_S$. We say that:*

- *A function $f \in C$ is s-***sparse** *if $L_0(f) \leq s$, ie., f has at most s nonzero entries in its polynomial expansion.*
- *f is (ε, d)-***concentrated** *if $\mathbb{E}[(f - \sum_{S,|S|\leq d} \hat{f}(S)\chi_S)^2] \geq 1 - \mathbf{E}$.*
- *C is $(\mathbf{E}, d, s)$-***bounded** *if for every $f \in C$, f is (ε, d)-concentrated and in addition C has L_1 norm bounded by s, that is, for every $f \in C$ we have $\sum_S |\hat{f}(S)| \leq s$.*

It is easy to see that the class of functions with bounded L_1 norm is more general than sparse functions. For example, the Boolean AND function has L_1 norm bounded by 1 but is not sparse.

We also have the following simple fact:

Fact 6.2.4. *[90] Let f be such that $L_1(f) \leq s$. Then there exists g such that g is $s^2/\mathbf{E}$ sparse and $E[(f-g)^2] \leq \varepsilon$. The function g is constructed by taking all coefficients of magnitude $\mathbf{E}/s$ or larger in f's expansion as a polynomial.*

6.2.2 Compressed sensing and sparse recovery

In the problem of *sparse recovery*, a learner attempts to recover a sparse vector $x \in \mathbb{R}^n$ which is s sparse, i.e. $\|x\|_0 \leq s$, from an observation vector $y \in R^m$ that is assumed to equal

$$y = Ax + e,$$

where e is assumed to be zero-mean, usually Gaussian, noise. The seminal work of [19, 30] shows how x can be recovered exactly under various conditions on the observation matrix $A \in \mathbb{R}^{m \times n}$ and the noise. The usual method for recovering the signal proceeds by solving a convex optimization problem consisting of ℓ_1 minimization as follows (for some parameter $\lambda > 0$):

$$\min_{x \in \mathbb{R}^n} \left\{ \|x\|_1 + \lambda \|Ax - y\|_2^2 \right\}. \tag{6.1}$$

The above formulation comes in many equivalent forms (e.g., Lasso), where one of the objective parts may appear as a hard constraint.

For our work, the most relevant extension of traditional sparse recovery is due to Rauhut [103], who considers the problem of sparse recovery when the measurements are evaluated according to a *random orthonormal family*. More concretely, fix $x \in \mathbb{R}^n$ with s non-zero entries. For K-bounded random orthonormal family $\mathcal{F} = \{\psi_1, \ldots, \psi_N\}$, and m independent draws $z^1, \ldots, z^m$ from corresponding distribution $\mathcal{D}$ define the $m \times N$ matrix A such that $A_{ij} = \psi_j(z^i)$. Rauhut gives the following result for recovering sparse vectors x:

Theorem 6.2.5 (Sparse Recovery for Random Orthonormal Families, [103] Theorem 4.4)**.** *Given as input matrix $A \in \mathbb{R}^{m \times N}$ and vector y with $y_i = Ax + e_i$ for some vector e with $\|e\|_2 \leq \eta\sqrt{m}$, mathematical program* (6.1) *finds a vector x^* such that (for constants c_1 and c_2)*

$$\|x - x^*\|_2 \leq c_1 \frac{\sigma_s(x)_1}{\sqrt{s}} + c_2\eta$$

with probability $1 - \delta$ as long as, for sufficiently large constant C,

$$m \geq CK^2 \log K \cdot s \log^3 s \cdot \log^2 N \cdot \log(1/\delta).$$

The term $\sigma_s(x)_1$ is equal to $\min\{\|x - z\|_1, z \text{ is } s \text{ sparse}\}$. Recent work [16, 48] has improved the dependence on the polylog factors in the lower bound for m.

6.3 Basic algorithm and main theoretical results

The main component of our spectral algorithm for hyperparameter optimization is given in Algorithm 3. It is essentially an extension of sparse recovery (basis pursuit or Lasso) to the orthogonal basis of polynomials in addition to an optimization step. See Figure 6.1 for an illustration. We prove Harmonica's theoretical guarantee, and show how it gives rise to new theoretical results in learning from the uniform distribution.

In the next section we describe extensions of this basic algorithm to a more practical algorithm with various heuristics to improve its performance.

Theorem 6.3.1 (Noiseless recovery)**.** *Let $\{\psi_S\}$ be a K-bounded orthonormal polynomial basis for distribution $\mathcal{D}$. Let $f : \mathbb{R}^n \mapsto \mathbb{R}$ be a $(0, d, s)$-bounded function as per definition 6.2.3 with respect to the basis ψ_S. Then Algorithm 3, in time $n^{O(d)}$ and sample complexity $T = \tilde{O}(K^2 s \cdot d \log n)$, returns*

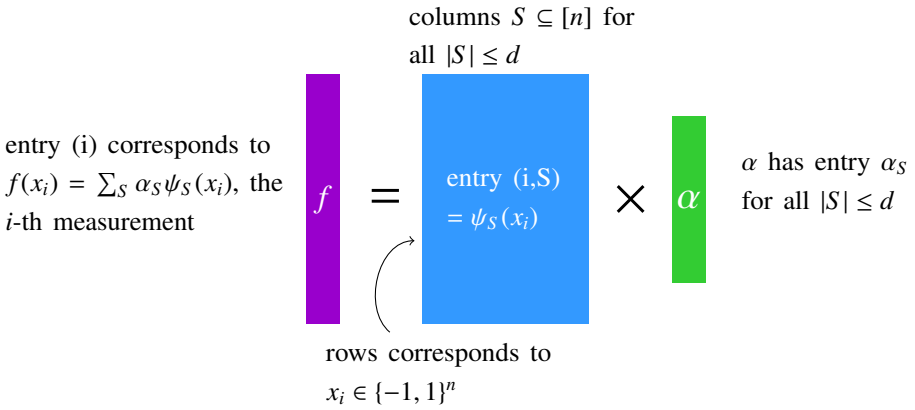


Figure 6.1: Compressed sensing over the Fourier domain: Harmonica recovers the Fourier coefficients of a sparse low degree polynomial $\sum_S \alpha_S \Psi_S(x_i)$ from observations $f(x_i)$ of randomly chosen points $x_i \in \{-1,1\}^n$.

Algorithm 3 Harmonica-1

1: Input: oracle for f, number of samples T, sparsity s, degree d, parameter λ.
2: Invoke PSR(f, T, s, d, λ) (Procedure 4) to obtain (g, J), where g is a function defined on variables specified by index set $J \subseteq [n]$.
3: Set the variables in $[n] \setminus J$ to arbitrary values, compute a minimizer $x^\star \in \arg\min g_i(x)$.
4: **return** $x^\star$

Procedure 4 Polynomial Sparse Recovery (PSR)

1: Input: oracle for f, number of samples T, sparsity s, degree d, regularization parameter λ
2: Query T random samples: $\{f(x_1), ..., f(x_T)\}$.
3: Solve sparse d-polynomial regression over all polynomials up to degree d

$$\underset{\alpha \in \mathbb{R}^{\binom{n}{d}}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{T} \left(\sum_{|S| \le d} \alpha_S \psi_S(x_i) - f(x_i) \right)^2 + \lambda \|\alpha\|_1 \right\} \tag{6.2}$$

4: Let $S_1, ..., S_s$ be the indices of the largest coefficients of $\vec{\alpha}$. Let g be the polynomial

$$g(x) = \sum_{i \in [s]} \alpha_{S_i} \psi_{S_i}(x)$$

5: **return** g and $J = \cup_{i=1}^{s} S_i$

$x^\star$ such that

$$x^\star \in \operatorname{argmin} f(x)$$

This theorem, and indeed most of the theoretical results of this paper, follow from the main recovery properties of Procedure 4. Our main technical lemma follows the same outline of the compressed sensing result due to Stobbe and Krause[1] [120] but with a generalization to functions that are *approximately* sparse and low-degree:

Lemma 6.3.2 (Noisy recovery)**.** *Let $\{\psi_S\}$ be a K-bounded orthonormal polynomial basis for distribution $\mathcal{D}$. Let $f : \mathbb{R}^n \mapsto \mathbb{R}$ be a $(\mathbf{E}/4, d, s)$-bounded as per definition 6.2.3 with respect to the basis ψ_S. Then Procedure 4 finds a function $g \equiv_\varepsilon f$ in time $O(n^d)$ and sample complexity $T = \tilde{O}(K^2 s^2/\varepsilon \cdot d \log n)$.*

In the rest of this section we proceed to prove the main lemma and derive the theorem. Recall the Chebyshev inequality:

Fact 6.3.3 (Multidimensional Chebyshev inequality)**.** *Let X be an m dimensional random vector, with expected value $\mu = \mathbb{E}[X]$, and covariance matrix $V = \mathbb{E}[(X-\mu)(X-\mu)^T]$.*

If V is a positive definite matrix, for any real number $\delta > 0$:

$$\Pr(\sqrt{(X-\mu)^T V^{-1}(X-\mu)} > \delta) \leq \frac{m}{\delta^2}$$

Proof of Lemma 6.3.2. For ease of notation we assume $K = 1$. Let f be an $(\varepsilon/4, s, d)$-bounded function written in the orthonormal basis as $\sum_S \hat{f}(S)\psi_S$. We can equivalently write f as $f = h + g$, where h is a degree d polynomial that only includes coefficients of magnitude at least $\mathbf{E}/4s$ and the constant term of the polynomial expansion of f.

Since $L_1(f) = \sum_S |\hat{f_S}| \leq s$, by Fact 6.2.4 we have that h is $4s^2/\mathbf{E} + 1$ sparse. The function g is thus the sum of the remaining $\hat{f}(S)\psi_S$ terms not included in h.

[1]Thanks to Vitaly Feldman for pointing it out.

Draw m (to be chosen later) random labeled examples $\{(z^1, y^1), \ldots, (z^m, y^m)\}$ and enumerate all $N = n^d$ basis functions ψ_S for $|S| \leq d$ as $\{\psi_1, \ldots, \psi_N\}$. Form matrix A such that $A_{ij} = \psi_j(z^i)$ and consider the problem of recovering $4s^2/\mathbf{E} + 1$ sparse x given $Ax + e = y$ where x is the vector of coefficients of h, the ith entry of y equals y^i, and $e_i = g(z^i)$.

We will prove that with constant probability over the choice m random examples, $\|e\|_2 \leq \sqrt{\varepsilon m}$. Applying Theorem 6.2.5 by setting $\eta = \sqrt{\varepsilon}$ and observing that $\sigma_{4s^2/\varepsilon+1}(x)_1 = 0$, we will recover x' such that $\|x - x'\|_2^2 \leq c_2^2\varepsilon$ for some constant c_2. As such, for the function $\tilde{f} = \sum_{i=1}^N x'_i\psi_i$ we will have $\mathbb{E}[\|h - \tilde{f}\|^2] \leq c_2^2\varepsilon$ by Parseval's identity. Note, however, that we may rescale ε by constant factor $1/(2c_2^2)$ to obtain error $\varepsilon/2$ and only incur an additional constant (multiplicative) factor in the sample complexity bound.

By the definition of g, we have

$$\|g\|^2 = \left(\sum_{S,|S|>d} \hat{f}(S)^2 + \sum_R \hat{f}(R)^2 \right) \tag{6.3}$$

where each $\hat{f}(R)$ is of magnitude at most $\mathbf{E}/4s$. By Fact 6.2.4 and Parseval's identity we have $\sum_R \hat{f}(R)^2 \leq \varepsilon/4$. Since f is $(\varepsilon/4, d)$-concentrated we have $\sum_{S,|S|>d} \hat{f}(S)^2 \leq \varepsilon/4$. Thus, $\|g\|^2$ is at most $\varepsilon/2$. Therefore, by triangle inequality $\mathbb{E}[\|f - \tilde{f}\|^2] \leq \mathbb{E}[\|h - \tilde{f}\|^2] + \mathbb{E}[\|g\|^2] \leq \varepsilon$.

It remains to bound $\|e\|_2$. Note that since the examples are chosen independently, the entries $e_i = g(z^i)$ are independent random variables. Since g is a linear combination of orthonormal monomials (not including the constant term), we have $\mathbb{E}_{z\sim D}[g(z)] = 0$. Here we can apply linearity of variance (the covariance of ψ_i and ψ_j is zero for all $i \neq j$) and calculate the variance

$$\mathrm{Var}(g(z^i)) = (\sum_{S,|S|>d} \hat{f}(S)^2 + \sum_R \hat{f}(R)^2)$$

With the same calculation as (6.3), we know $\mathrm{Var}(g(z^i))$ is at most $\varepsilon/2$.

Now consider the covariance matrix V of the vector e which equals $\mathbb{E}[ee^\top]$ (recall every entry of e has mean 0). Then V is a diagonal matrix (covariance between two independent samples is zero), and every diagonal entry is at most $\mathbf{E}/2$. Applying Fact 6.3.3 we have

$$\Pr(\|e\|_2 > \sqrt{\frac{\varepsilon}{2}}\delta) \leq \frac{m}{\delta^2}.$$

Setting $\delta = \sqrt{2m}$, we conclude that $\Pr(\|e\|_2 > \sqrt{\mathbf{E}m}) \leq \frac{1}{2}$. Hence with probability at least $1/2$, we have that $\|e\|_2 \leq \sqrt{\varepsilon m}$. From Theorem 6.2.5, we may choose $m = \tilde{O}(s^2/\varepsilon \cdot \log n^d)$. This completes the proof. Note that the probability $1/2$ above can be boosted to any constant probability with a constant factor loss in sample complexity. □

Remark: Note that the above proof also holds in the *adversarial* or *agnostic* noise setting. That is, an adversary could add a noise vector v to the labels received by the learner. In this case, the learner will see label vector $y = Ax + e + v$. If $\|v\|_2 \leq \sqrt{\gamma m}$, then we will recover a polynomial with squared-error at most $\mathbf{E} + O(\gamma)$ via re-scaling ε by a constant factor and applying the triangle inequality to $\|e + v\|_2$.

While this noisy recovery lemma is the basis for our enhanced algorithm in the next section as well as the learning-theoretic result on learning of decision trees detailed in the next subsection, it does not imply recovery of the global optimum. The reason is that noisy recovery guarantees that we output a hypothesis *close* to the underlying function, but even a single noisy point can completely change the optimum.

Nevertheless, we can use our techniques to prove recovery of optimality for functions that are computed *exactly* by a sparse, low-degree polynomial.

Proof of Theorem 6.3.1. There are at most $N = n^d$ polynomials ψ_S with $|S| \leq d$. Let the enumeration of these polynomials be $\psi_1, \ldots, \psi_N$. Draw m labeled examples $\{(z^1, y^1), \ldots, (z^m, y^m)\}$ independently from $\mathcal{D}$ and construct an $m \times N$ matrix A with $A_{ij} = \psi_j(z^i)$. Since f can be written as an s sparse linear combination of $\psi_1, \ldots, \psi_N$, there exists an s-sparse vector x such that $Ax = y$ where the ith entry of y is y^i. Hence we can apply Theorem 6.2.5 to recover x exactly. These are the s non-zero coefficients of f's expansion in terms of $\{\psi_S\}$. Since f is recovered exactly, its minimizer is found in the optimization step. □

6.3.1 Application: learning decision trees

Here we observe that our results imply new bounds for decision-tree learning. For example, we obtain the first quasi-polynomial time algorithm for learning decision trees with respect to the uniform distribution on $\{-1, 1\}^n$ with polynomial sample complexity and an optimal dependence on the error parameter ε:

Corollary 6.3.4. *Let $\mathcal{X} = \{-1, 1\}^n$ and let C be the class of all decision trees of size s on n variables. Then C is learnable with respect to the uniform distribution in time $n^{O(\log(s/\varepsilon))}$ and sample complexity $m = \tilde{O}(s^2/\varepsilon \cdot \log n)$. Further, if the labels are corrupted by arbitrary noise vector v such that $\|v\|_2 \leq \sqrt{\gamma m}$, then the output classifier will have squared-error at most $\varepsilon + O(\gamma)$.*

Proof. As mentioned earlier, the orthonormal polynomial basis for the class of Boolean functions with respect to the uniform distribution on $\{-1, 1\}^n$ is the class of parity functions $\{\chi_S\}$ for $S \subseteq \{-1, 1\}^n$. Further, it is easy to show that for Boolean function f, if $\mathbb{E}[(h - f)^2] \leq \varepsilon$ then $\Pr[\mathsf{sign}(h(x)) \neq f(x)] \leq \varepsilon$. The corollary now follows by applying Lemma 6.3.2 and two known structural facts about decision trees: 1) a tree of size s is $(\mathbf{E}, \log(s/\mathbf{E}))$-concentrated and has L_1 norm bounded by s (see e.g., Mansour [90]) and 2) by Fact 6.2.4, for any function f with L_1

norm bounded by s (i.e., a decision tree of size s), there exists an $s^2/\mathbf{E}$ sparse function g such that $\mathbb{E}[(f-g)^2] \leq \varepsilon$. The noise tolerance property follows immediately from the remark after the proof of Lemma 6.3.2. □

Comparison with the "Low-Degree" Algorithm. Prior work for learning decision trees (more generally Boolean functions that are approximated by low-degree polynomials) used the celebrated "low-degree" algorithm of Linial, Mansour, and Nisan [83]. Their algorithm uses random sampling to estimate each low-degree Fourier coefficient to high accuracy. In contrast, the compressed-sensing approach of Stobbe and Krause [120] takes advantage of the incoherence of the design matrix and gives results that seem unattainable from the "low-degree" algorithm.

For learning noiseless, Boolean decision trees, the low-degree algorithm uses quasipolynomial time and sample complexity $\tilde{O}(s^2/\mathbf{E}^2 \cdot \log n)$ to learn to accuracy ε. It is not clear, however, how to obtain any noise tolerance from their approach.

For general real-valued decision trees where B is an upper bound on the maximum value at any leaf of a size s tree, our algorithm will succeed with sample complexity $\tilde{O}(B^2 s^2/\varepsilon \cdot \log n)$ and be tolerant to noise while the low-degree algorithm will use $\tilde{O}(B^4 s^2/\varepsilon^2 \cdot \log n)$ (and will have no noise tolerance properties). Note the improvement in the dependence on ε (even in the noiseless setting), which is a consequence of the RIP property of the random orthonormal family.

6.4 Harmonica: the full algorithm

Rather than applying Algorithm 3 directly, we found that performance is greatly enhanced by iteratively using Procedure 4 to estimate the most influential hyperparameters and their optimal values.

In the rest of this section we describe this iterative heuristic, which essentially runs Algorithm 3 for multiple stages. More concretely, we continue to invoke the PSR subroutine until the search space becomes small enough for us to use a "base" hyperparameter optimizer (in our case either SH or Random Search).

The space of minimizing assignments to a multivariate polynomial is a highly non-convex set that may contain many distinct points. As such, we take an average of several of the best minimizers (of subsets of hyperparameters) during each stage.

In order to describe this formally we need the following definition of a restriction of function:

Definition 6.4.1 (restriction [96])**.** *Let $f \in \{-1,1\}^n \mapsto \mathbb{R}$, $J \subseteq [n]$, and $z \in \{-1,1\}^J$ be given. We call (J,z) a restriction pair of function f. We denote $f_{J,z}$ the function over $n - |J|$ variables given by setting the variables of J to z.*

We can now describe our main algorithm (Algorithm 5). Here q is the number of stages for which we apply the PSR subroutine, and the restriction size t serves as a tie-breaking rule for the best minimizers (which can be set to 1).

Algorithm 5 Harmonica-q

1: Input: oracle for f, number of samples T, sparsity s, degree d, regularization parameter λ, number of stages q, restriction size t, base hyperparameter optimizer ALG.
2: **for** stage $i = 1$ to q **do**
3: Invoke PSR(f, T, s, d, λ) (Procedure 4) to obtain (g_i, J_i), where g_i is a function defined on variables specified by index set $J_i \subseteq [n]$.
4: Let $M_i = \{x_1^\star, ..., x_t^\star\} = \arg\min g_i(x)$ be the best t minimizers of g_i.
5: Let $f_i = \mathbb{E}_{k\in[t]}[f_{J_i,x_k^\star}]$ be the expected restriction of f according to minimizers M_i.[2]
6: Set $f = f_i$.
7: **return** Search for the global minimizer of f_q using base optimizer ALG

6.4.1 Algorithm attributes and heuristics

Scalability. If the hidden function if s-sparse, Harmonica can find such a sparse function using $\tilde{O}(s \log s)$ samples. If at every stage of Harmonica, the target function can be approximated by an s sparse function, we only need $\tilde{O}(qs \log s)$ samples where q is the number of stages. For real world applications such as deep neural network hyperparameter tuning, it seems (empirically) reasonable to assume that the hidden function is indeed sparse at every stage (see Section 6.5).

For Hyperband [81], SH [65] or Random Search, even if the function is s-sparse, in order to cover the optimal configuration by random sampling, we need $\Omega(2^s)$ samples.

Optimization time. Harmonica runs the Lasso [125] algorithm after each stage to solve (6.2), which is a well studied convex optimization problem and has very fast implementations. Hyperband and SH are also efficient in terms of running time as a function of the number of function evaluations, and require sorting or other simple computations. The running time of Bayesian optimization is cubic in number of function evaluations, which limits applicability for large number of evaluations / high dimensionality.

Parallelizability. Harmonica, similar to Hyperband, SH, and Random Search, has straightforward parallel implementations. In every stage of those algorithms, we could simply evaluate the objective functions over randomly chosen points in parallel.

It is hard to run Bayesian optimization algorithm in parallel due to its inherent serial nature. Previous works explored variants in which multiple points are evaluated at the same time in parallel [128], though speed ups do not grow linearly in the number of machines, and the batch size is usually limited to a small number.

[2] In order to evaluate f_i, we first sample $k \in [t]$ to obtain f_{J_i, x_k^*}, and then evaluate f_{J_i, x_k^*}.

Feature Extraction. Harmonica is able to extract important features with weights in each stages, which automatically sorts all the features according to their importance. See Section C.1.2.

6.5 Experiments with training deep networks

We compare Harmonica[3] with Spearmint[4] [117], Hyperband, SH[5] and Random Search. Both Spearmint and Hyperband are state-of-the-art algorithms, and it is observed that Random Search 2x (Random Search with doubled function evaluation resources) is a very competitive benchmark that beats many algorithms[6].

Our first experiment is over training residual network on Cifar-10 dataset[7]. We included 39 binary hyperparameters, including initialization, optimization method, learning rate schedule, momentum rate, etc. Table C.1 (Section C.1.1) details the hyperparameters considered. We also include 21 dummy variables to make the task more challenging. Notice that Hyperband, SH, and Random Search are agnostic to the dummy variables in the sense that they just set the value of dummy variables randomly, therefore select essentially the same set of configurations with or without the dummy variables. Only Harmonica and Spearmint are sensitive to the dummy variables as they try to learn the high dimensional function space. To make a fair comparison, we run Spearmint without any dummy variables.

As most hyperparameters have a consistent effect as the network becomes deeper, a common hand-tuning strategy is "tune on small network, then apply the knowledge to big network" (See discussion in Section C.1.3). Harmonica can also exploit this strategy as it selects important features

[3] A python implementation of Harmonica can be found at `https://github.com/callowbird/Harmonica`
[4] `https://github.com/HIPS/Spearmint.git`
[5] We implemented a parallel version of Hyperband and SH in Lua.
[6] E.g., see [104, 105].
[7] `https://github.com/facebook/fb.resnet.torch`

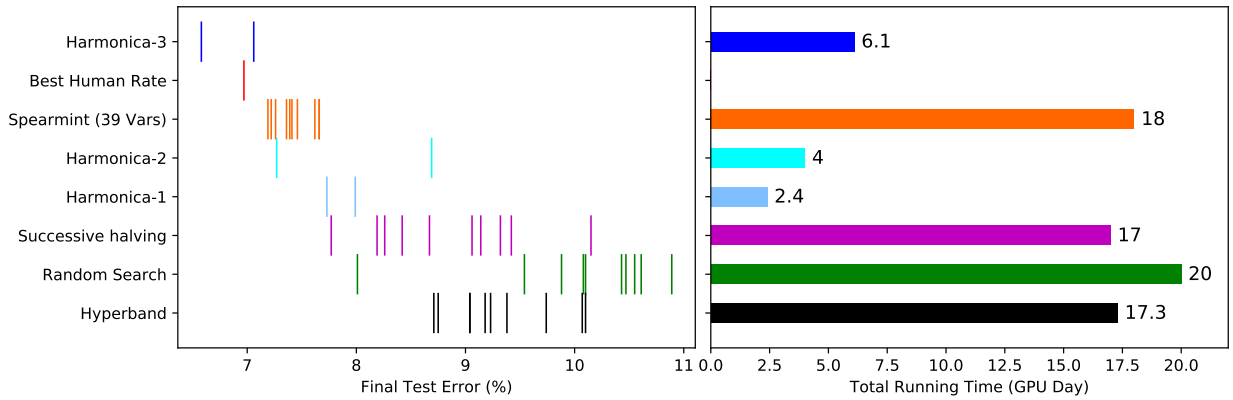


Figure 6.2: Distribution of the best results and running time of different algorithms

stage-by-stage. More specifically, during the feature selection stages, we run Harmonica for tuning an 8 layer neural network with 30 training epochs. At each stage, we take 300 samples to extract 5 important features, and set restriction size $t = 4$ (see Procedure 4). After that, we fix all the important features, and run the SH or Random Search as our base algorithm on the big 56 layer neural network for training the whole 160 epochs[8]. To clarify, "stage" means the stages of the hyperparameter algorithms, while "epoch" means the epochs for training the neural network.

6.5.1 Performance

We tried three versions of Harmonica for this experiment, Harmonica with 1 stage (Harmonica-1), 2 stages (Harmonica-2) and 3 stages (Harmonica-3). All of them use SH as the base algorithm. The top 10 test error results and running times of the different algorithms are depicted in Figure 6.2. SH based algorithms may return fewer than 10 results. For more runs of variants of Harmonica and its resulting test error, see Figure 6.3 (the results are similar to Figure 6.2).

[8]Other algorithms like Spearmint, Hyperband, etc. can be used as the base algorithms as well.

Test error and scalability: Harmonica-1 uses less than $1/7$ time of Hyperband and $1/8$ time compared with Random Search, but gets better results than the competing algorithms. It beats the Random Search 8x benchmark (stronger than Random Search 2x benchmark of [81]). Harmonica-2 uses slightly more time, but is able to find better results, which are comparable with Spearmint with $4.5\times$ running time.

Improving upon human-tuned parameters: Harmonica-3 obtains a better test error (6.58%) as compared to the best hand-tuning rate 6.97% reported in [51][9]. Harmonica-3 uses only 6.1 GPU days, which is less than half day in our environment, as we have 20 GPUs running in parallel. Notice that we did not cherry pick the results for Harmonica-3. In Section 6.5.3 we show that by running Harmonica-3 for longer time, one can obtain a few other solutions better than hand tuning.

Performance of provable methods: Harmonica-1 has noiseless and noisy recovery guarantees (Lemma 6.3.2), which are validated experimentally.

6.5.2 Average test error for each stage

We computed the average test error among 300 random samples for an 8 layer network with 30 epochs after each stage. See Figure 6.4. After selecting 5 features in stage 1, the average test error drops from 60.16 to 33.3, which indicates the top 5 features are very important. As we proceed to stage 3, the improvement on test error becomes less significant as the selected features at stage 3 have mild contributions.

[9] 6.97% is the rate obtained by residual network, and there are new network structures like wide residual network [130] or densenet [57] that achieve better rates for Cifar-10.

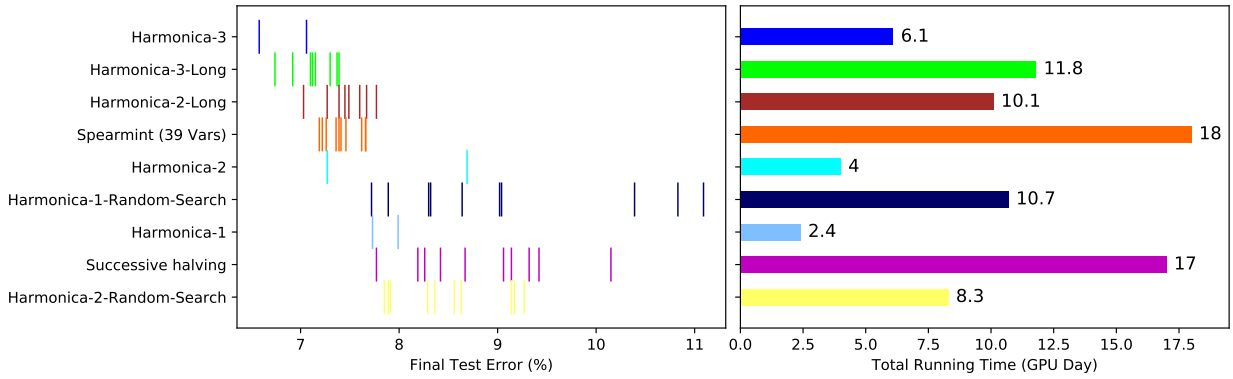


Figure 6.3: Comparing different variants of Harmonica with SH on test error and running time

6.5.3 Hyperparameters for Harmonica

To be clear, Harmonica itself has six hyperparameters that one needs to set including the number of stages, ℓ_1 regularizer for Lasso, the number of features selected per stage, base algorithm, small network configuration, and the number of samples per stage. Note, however, that we have reduced the search space of general hyperparameter optimization down to a set of only six hyperparameters. Empirically, our algorithm is robust to different settings of these parameters, and we did not even attempt to tune some of them (e.g., small network configuration).

Base algorithm and #stages. We tried different versions of Harmonica, including Harmonica with 1 stage, 2 stages and 3 stages using SH as the base algorithm (Harmonica-1, Harmonica-2, Harmonica-3), with 1 stage and 2 stages using Random Search as the base algorithm (Harmonica-1-Random-Search, Harmonica-2-Random-Search), and with 2 stages and 3 stages running SH as the base for longer time (Harmonica-2-Long, Harmonica-3-Long). As can be seen in Figure 6.3, most variants produce better results than SH and use less running time. Moreover, if we run Harmonica for longer time, we will obtain more stable solutions with less variance in test error.

Table 6.1: Stable ranges for parameters in Lasso

Parameter	Stage 1	Stage 2	Stage 3
λ	$[0.01, 4.5]$	$[0.1, 2.5]$	$[0.5, 1.1]$
#Samples	≥ 250	≥ 180	≥ 150

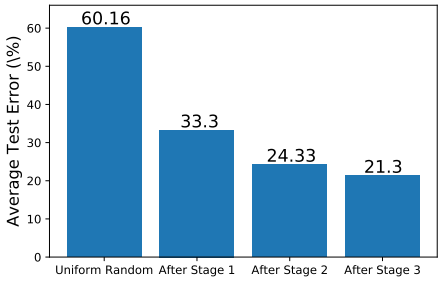


Figure 6.4: Average test error drops.

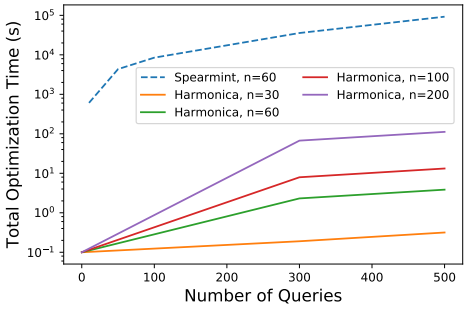


Figure 6.5: Optimization time comparison

Lasso parameters are stable. See Table 6.1 for stable range for regularization term λ and the number of samples. Here stable range means as long as the parameters are set in this range, the top 5 features and the signs of their weights (which are what we need for computing $g(x)$ in Procedure 4) do not change. In other words, the feature selection outcome is not affected. When parameters are outside the stable ranges, usually the top features are still unchanged, and we miss only one or two out of the five features.

On the degree of features. We set degree to be three because it does not find any important features with degree larger than this. Since Lasso can be solved efficiently (less than 5 minutes in our experiments), the choice of degree can be decided automatically.

CHAPTER 7

CONCLUSION

In this dissertation, we have presented results on analysis of SGD for general functions and two layer neural networks, as well as a new hyperparameter tuning algorithm using compressed sensing in Fourier domain.

There are many exciting open problems. For example, can we identify strict saddle properties for other important machine learning problems? Can we find more general conditions for SGD to escape bad local minima? Can we understand sufficient conditions for SGD to find flat local minima, and what are the generalization properties of flat local minima, compared with sharp ones? Can we get better characterization of loss surfaces of different machine learning problems, and can we improve these loss surface by modifying the loss function or the problem structures? Can we provide similar convergence guarantee or dynamic analysis of SGD for reinforcement learning?

It is also important to understand the convergence behavior of SGD for deep neural networks, which might need completely different techniques from what we are using for two layer network, as directly computing the gradients in the analytic form is infeasible for deep networks. Moreover, empirically it seems that the local minima of deep neural networks are almost equally good, and it would be great to prove it rigorously.

For hyperparameter tuning, Harmonica can be further improved empirically or theoretically. For example, neural networks with proper regularizers might fit the black box functions better compared with compressed sensing, and therefore could work better in practice. From a theoretical perspective, we could use the learned decision tree as a tree metric for the target hyperparameter space, and apply multi-armed bandit based sampling to refine the tree metric, which could give us better sample complexity guarantees.

APPENDIX A

APPENDIX FOR ESCAPING SADDLE POINT

A.1 Detailed analysis for Section 3.2 in unconstrained case

In this section we give detailed analysis for noisy gradient descent, under the assumption that the unconstrained problem satisfies $(\lambda, \gamma, \varepsilon, \delta)$-strict saddle property.

The algorithm we investigate in Algorithm 1, we can combine the randomness in the stochastic gradient oracle and the artificial noise, and rewrite the update equation in form:

$$w_t = w_{t-1} - \eta(\nabla f(w_{t-1}) + \xi_{t-1}) \tag{A.1}$$

where η is step size, $\xi = SG(w_{t-1}) - \nabla f(w_{t-1}) + n$ (recall n is a random vector on unit sphere) is the combination of two source of noise.

By assumption, we know ξ's are independent and they satisfying $\mathbb{E}\xi = 0$, $\|\xi\| \leq Q + 1$. Due to the explicitly added noise in Algorithm 1, we further have $\mathbb{E}\xi\xi^T \succ \frac{1}{d}I$. For simplicity, we assume $\mathbb{E}\xi\xi^T = \sigma^2 I$, for some constant $\sigma = \tilde{\Theta}(1)$, then the algorithm we are running is exactly the same as Stochastic Gradient Descent (SGD). Our proof can be very easily extended to the case when $\frac{1}{d}I \preceq \mathbb{E}[\xi\xi^T] \preceq (Q + \frac{1}{d})I$ because both the upper and lower bounds are $\tilde{\Theta}(1)$.

We first restate the main theorem in the context of stochastic gradient descent.

Theorem A.1.1 (Main Theorem)**.** *Suppose a function $f(w) : \mathbb{R}^d \to \mathbb{R}$ that is $(\lambda, \gamma, \varepsilon, \delta)$-strict saddle, and has a stochastic gradient oracle where the noise satisfy $\mathbb{E}\xi\xi^T = \sigma^2 I$. Further, suppose the function is bounded by $|f(w)| \leq B$, is L-smooth and has ρ-Lipschitz Hessian. Then there exists a threshold $\eta_{\max} = \tilde{\Theta}(1)$, so that for any $\zeta > 0$, and for any $\eta \leq \eta_{\max}/\max\{1, \log(1/\zeta)\}$,*

with probability at least $1-\zeta$ *in* $t = \tilde{O}(\eta^{-2}\log(1/\zeta))$ *iterations, SGD outputs a point* w_t *that is* $\tilde{O}(\sqrt{\eta\log(1/\eta\zeta)})$*-close to some local minimum* $w^\star$*.*

Recall that $\tilde{O}(\cdot)$ ($\tilde{\Omega}, \tilde{\Theta}$) hides the factor that is polynomially dependent on all other parameters, but independent of η and ζ. So it focuses on the dependency on η and ζ. Throughout the proof, we interchangeably use both $\mathcal{H}(w)$ and $\nabla^2 f(w)$ to represent the Hessian matrix of $f(w)$.

As we discussed in the proof sketch in Section 3.2, we analyze the behavior of the algorithm in three different cases. The first case is when the gradient is large.

Lemma A.1.2. *Under the assumptions of Theorem A.1.1, for any point with* $\|\nabla f(w_0)\| \geq \sqrt{2\eta\sigma^2 Ld}$ *where* $\sqrt{2\eta\sigma^2 Ld} < \varepsilon$*, after one iteration we have:*

$$\mathbb{E}f(w_1) - f(w_0) \leq -\tilde{\Omega}(\eta^2) \tag{A.2}$$

Proof. Choose $\eta_{\max} < \frac{1}{L}$, then by update equation Eq.(A.1), we have:

$$\begin{aligned}
\mathbb{E}f(w_1) - f(w_0) &\leq \nabla f(w_0)^T \mathbb{E}(w_1 - w_0) + \frac{L}{2}\mathbb{E}\|w_1 - w_0\|^2 \\
&= \nabla f(w_0)^T \mathbb{E}\left(-\eta(\nabla f(w_0) + \xi_0)\right) + \frac{L}{2}\mathbb{E}\left\|-\eta(\nabla f(w_0) + \xi_0)\right\|^2 \\
&= -(\eta - \frac{L\eta^2}{2})\|\nabla f(w_0)\|^2 + \frac{\eta^2\sigma^2 Ld}{2} \\
&\leq -\frac{\eta}{2}\|\nabla f(w_0)\|^2 + \frac{\eta^2\sigma^2 Ld}{2} \leq -\frac{\eta^2\sigma^2 Ld}{2}
\end{aligned} \tag{A.3}$$

which finishes the proof. □

Lemma A.1.3. *Under the assumptions of Theorem A.1.1, for any initial point* w_0 *that is* $\tilde{O}(\sqrt{\eta}) < \delta$ *close to a local minimum* $w^\star$*, with probability at least* $1 - \zeta/2$*, we have following holds simultaneously:*

$$\forall t \leq \tilde{O}(\frac{1}{\eta^2}\log\frac{1}{\zeta}), \quad \|w_t - w^\star\| \leq \tilde{O}(\sqrt{\eta\log\frac{1}{\eta\zeta}}) < \delta \tag{A.4}$$

where $w^\star$ *is the locally optimal point.*

Proof. We shall construct a supermartingale and use Azuma's inequality [7] to prove this result.

Let filtration $\mathfrak{F}_t = \sigma\{\xi_0, \cdots \xi_{t-1}\}$, and note $\sigma\{\Delta_0, \cdots, \Delta_t\} \subset \mathfrak{F}_t$, where $\sigma\{\cdot\}$ denotes the sigma field. Let event $\mathfrak{E}_t = \{\forall \tau \leq t, \|w_\tau - w^\star\| \leq \mu\sqrt{\eta \log \frac{1}{\eta\zeta}} < \delta\}$, where μ is independent of (η, ζ), and will be specified later. To ensure the correctness of proof, $\tilde{O}$ notation in this proof will never hide any dependence on μ. Clearly there's always a small enough choice of $\eta_{\max} = \tilde{\Theta}(1)$ to make $\mu\sqrt{\eta \log \frac{1}{\eta\zeta}} < \delta$ holds as long as $\eta \leq \eta_{\max}/\max\{1, \log(1/\zeta)\}$. Also note $\mathfrak{E}_t \subset \mathfrak{E}_{t-1}$, that is $1_{\mathfrak{E}_t} \leq 1_{\mathfrak{E}_{t-1}}$.

By Definition 3.2.2 of $(\lambda, \gamma, \varepsilon, \delta)$-strict saddle, we know f is locally λ-strongly convex in the 2δ-neighborhood of $w^\star$. Since $\nabla f(w^\star) = 0$, we have

$$\nabla f(w_t)^T (w_t - w^\star) 1_{\mathfrak{E}_t} \geq \lambda \|w_t - w^\star\|^2 1_{\mathfrak{E}_t} \tag{A.5}$$

Furthermore, with $\eta_{\max} < \frac{\lambda}{L^2}$, using L-smoothness, we have:

$$\begin{aligned}
\mathbb{E}[\|w_t - w^\star\|^2 1_{\mathfrak{E}_{t-1}} | \mathfrak{F}_{t-1}] =& \mathbb{E}[\|w_{t-1} - \eta(\nabla f(w_{t-1}) + \xi_{t-1}) - w^\star\|^2 | \mathfrak{F}_{t-1}] 1_{\mathfrak{E}_{t-1}} \\
=& \left[\|w_{t-1} - w^\star\|^2 - 2\eta \nabla f(w_{t-1})^T (w_{t-1} - w^\star) + \eta^2 \|\nabla f(w_{t-1})\|^2 + \eta^2\sigma^2\right] 1_{\mathfrak{E}_{t-1}} \\
\leq& [(1 - 2\eta\lambda + \eta^2 L^2)\|w_{t-1} - w^\star\|^2 + \eta^2\sigma^2] 1_{\mathfrak{E}_{t-1}} \\
\leq& [(1 - \eta\lambda)\|w_{t-1} - w^\star\|^2 + \eta^2\sigma^2] 1_{\mathfrak{E}_{t-1}}
\end{aligned} \tag{A.6}$$

Therefore, we have:

$$\left[\mathbb{E}[\|w_t - w^\star\|^2 | \mathfrak{F}_{t-1}] - \frac{\eta}{\lambda}\right] 1_{\mathfrak{E}_{t-1}} \leq (1 - \eta\lambda)\left[\|w_{t-1} - w^\star\|^2 - \frac{\eta}{\lambda}\right] 1_{\mathfrak{E}_{t-1}} \tag{A.7}$$

Then, let $G_t = (1 - \eta\lambda)^{-t}(\|w_t - w^\star\|^2 - \frac{\eta}{\lambda})$, we have:

$$\mathbb{E}[G_t 1_{\mathfrak{E}_{t-1}} | \mathfrak{F}_{t-1}] \leq G_{t-1} 1_{\mathfrak{E}_{t-1}} \leq G_{t-1} 1_{\mathfrak{E}_{t-2}} \tag{A.8}$$

which means $G_t 1_{\mathfrak{E}_{t-1}}$ is a supermartingale.

Therefore, with probability 1, we have:

$$\begin{aligned}
&|G_t 1_{\mathfrak{E}_{t-1}} - \mathbb{E}[G_t 1_{\mathfrak{E}_{t-1}} | \mathfrak{F}_{t-1}]| \\
\leq &(1-\eta\lambda)^{-t}[\ \|w_{t-1} - \eta\nabla f(w_{t-1}) - w^\star\| \cdot \eta\|\xi_{t-1}\| + \eta^2\|\xi_{t-1}\|^2 - \eta^2\sigma^2\ ]1_{\mathfrak{E}_{t-1}} \\
\leq &(1-\eta\lambda)^{-t} \cdot \tilde{O}(\mu\eta^{1.5}\log^{\frac{1}{2}}\frac{1}{\eta\zeta}) = d_t
\end{aligned} \tag{A.9}$$

Let

$$c_t = \sqrt{\sum_{\tau=1}^{t} d_\tau^2} = \tilde{O}(\mu\eta^{1.5}\log^{\frac{1}{2}}\frac{1}{\eta\zeta})\sqrt{\sum_{\tau=1}^{t}(1-\eta\lambda)^{-2\tau}} \tag{A.10}$$

By Azuma's inequality, with probability less than $\tilde{O}(\eta^3\zeta)$, we have:

$$G_t 1_{\mathfrak{E}_{t-1}} > \tilde{O}(1)c_t\log^{\frac{1}{2}}(\frac{1}{\eta\zeta}) + G_0 \tag{A.11}$$

We know $G_t > \tilde{O}(1)c_t\log^{\frac{1}{2}}(\frac{1}{\eta\zeta}) + G_0$ is equivalent to:

$$\|w_t - w^\star\|^2 > \tilde{O}(\eta) + \tilde{O}(1)(1-\eta\lambda)^t c_t\log^{\frac{1}{2}}(\frac{1}{\eta\zeta}) \tag{A.12}$$

We know:

$$\begin{aligned}
&(1-\eta\lambda)^t c_t\log^{\frac{1}{2}}(\frac{1}{\eta\zeta}) = \mu\cdot\tilde{O}(\eta^{1.5}\log\frac{1}{\eta\zeta})\sqrt{\sum_{\tau=1}^{t}(1-\eta\lambda)^{2(t-\tau)}} \\
=&\mu\cdot\tilde{O}(\eta^{1.5}\log\frac{1}{\eta\zeta})\sqrt{\sum_{\tau=0}^{t-1}(1-\eta\lambda)^{2\tau}} \leq \mu\cdot\tilde{O}(\eta^{1.5}\log\frac{1}{\eta\zeta})\sqrt{\frac{1}{1-(1-\eta\lambda)^2}} = \mu\cdot\tilde{O}(\eta\log\frac{1}{\eta\zeta})
\end{aligned} \tag{A.13}$$

This means Azuma's inequality implies, there exist some $\tilde{C} = \tilde{O}(1)$ so that:

$$P\left(\mathfrak{E}_{t-1}\cap\left\{\|w_t - w^\star\|^2 > \mu\cdot\tilde{C}\eta\log\frac{1}{\eta\zeta})\right\}\right) \leq \tilde{O}(\eta^3\zeta) \tag{A.14}$$

By choosing $\mu > \tilde{C}$, this is equivalent to:

$$P\left(\mathfrak{E}_{t-1}\cap\left\{\|w_t - w^\star\|^2 > \mu^2\eta\log\frac{1}{\eta\zeta}\right\}\right) \leq \tilde{O}(\eta^3\zeta) \tag{A.15}$$

Then we have:

$$P(\overline{\mathfrak{E}}_t) = P\left(\mathfrak{E}_{t-1} \cap \left\{\|w_t - w^\star\| > \mu\sqrt{\eta \log \frac{1}{\eta\zeta}}\right\}\right) + P(\overline{\mathfrak{E}}_{t-1}) \leq \tilde{O}(\eta^3\zeta) + P(\overline{\mathfrak{E}}_{t-1}) \tag{A.16}$$

By initialization conditions, we know $P(\overline{\mathfrak{E}}_0) = 0$, and thus $P(\overline{\mathfrak{E}}_t) \leq t\tilde{O}(\eta^3\zeta)$. Take $t = \tilde{O}(\frac{1}{\eta^2}\log\frac{1}{\zeta})$, we have $P(\overline{\mathfrak{E}}_t) \leq \tilde{O}(\eta\zeta\log\frac{1}{\zeta})$. When $\eta_{\max} = \tilde{O}(1)$ is chosen small enough, and $\eta \leq \eta_{\max}/\log(1/\zeta)$, this finishes the proof. □

Lemma A.1.4. *Under the assumptions of Theorem A.1.1, for any initial point w_0 where $\|\nabla f(w_0)\| \leq \sqrt{2\eta\sigma^2 Ld} < \varepsilon$, and $\lambda_{\min}(\mathcal{H}(w_0)) \leq -\gamma$, then there is a number of steps T that depends on w_0 such that:*

$$\mathbb{E}f(w_T) - f(w_0) \leq -\tilde{\Omega}(\eta) \tag{A.17}$$

The number of steps T has a fixed upper bound T_{max} that is independent of w_0 where $T \leq T_{max} = O((\log d)/\gamma\eta)$.

Remark 2. *In general, if we relax the assumption $\mathbb{E}\xi\xi^T = \sigma^2 I$ to $\sigma_{\min}^2 I \leq \mathbb{E}\xi\xi^T \leq \sigma_{\max}^2 I$, the upper bound T_{max} of number of steps required in Lemma A.1.4 would be increased to $T_{max} = O(\frac{1}{\gamma\eta}(\log d + \log\frac{\sigma_{\max}}{\sigma_{\min}}))$*

As we described in the proof sketch, the main idea is to consider a coupled update sequence that correspond to the local second-order approximation of $f(x)$ around w_0. We characterize this sequence of update in the next lemma.

Lemma A.1.5. *Under the assumptions of Theorem A.1.1. Let $\tilde{f}$ defined as local second-order approximation of $f(x)$ around w_0:*

$$\tilde{f}(w) \doteq f(w_0) + \nabla f(w_0)^T(w - w_0) + \frac{1}{2}(w - w_0)^T\mathcal{H}(w_0)(w - w_0) \tag{A.18}$$

$\{\tilde{w}_t\}$ be the corresponding sequence generated by running SGD on function $\tilde{f}$, with $\tilde{w}_0 = w_0$. For simplicity, denote $\mathcal{H} = \mathcal{H}(w_0) = \nabla^2 f(w_0)$, then we have analytically:

$$\nabla\tilde{f}(\tilde{w}_t) = (1 - \eta\mathcal{H})^t\nabla f(w_0) - \eta\mathcal{H}\sum_{\tau=0}^{t-1}(1 - \eta\mathcal{H})^{t-\tau-1}\xi_\tau \tag{A.19}$$

$$\tilde{w}_t - w_0 = -\eta \sum_{\tau=0}^{t-1} (1 - \eta\mathcal{H})^\tau \nabla f(w_0) - \eta \sum_{\tau=0}^{t-1} (1 - \eta\mathcal{H})^{t-\tau-1} \xi_\tau \tag{A.20}$$

Furthermore, for any initial point w_0 where $\|\nabla f(w_0)\| \le \tilde{O}(\eta) < \varepsilon$, and $\lambda_{\min}(\mathcal{H}(w_0)) = -\gamma_0$. Then, there exist a $T \in \mathbb{N}$ satisfying:

$$\frac{d}{\eta\gamma_0} \le \sum_{\tau=0}^{T-1} (1 + \eta\gamma_0)^{2\tau} < \frac{3d}{\eta\gamma_0} \tag{A.21}$$

with probability at least $1 - \tilde{O}(\eta^3)$, we have following holds simultaneously for all $t \le T$:

$$\|\tilde{w}_t - w_0\| \le \tilde{O}(\eta^{\frac{1}{2}} \log \frac{1}{\eta}); \qquad \|\nabla \tilde{f}(\tilde{w}_t)\| \le \tilde{O}(\eta^{\frac{1}{2}} \log \frac{1}{\eta}) \tag{A.22}$$

Proof. Denote $\mathcal{H} = \mathcal{H}(w_0)$, since $\tilde{f}$ is quadratic, clearly we have:

$$\nabla \tilde{f}(\tilde{w}_t) = \nabla \tilde{f}(\tilde{w}_{t-1}) + \mathcal{H}(\tilde{w}_t - \tilde{w}_{t-1}) \tag{A.23}$$

Substitute the update equation of SGD in Eq.(A.23), we have:

$$\begin{aligned}
\nabla \tilde{f}(\tilde{w}_t) &= \nabla \tilde{f}(\tilde{w}_{t-1}) - \eta\mathcal{H}(\nabla \tilde{f}(\tilde{w}_{t-1}) + \xi_{t-1}) \\
&= (1 - \eta\mathcal{H}) \nabla \tilde{f}(\tilde{w}_{t-1}) - \eta\mathcal{H}\xi_{t-1} \\
&= (1 - \eta\mathcal{H})^2 \nabla \tilde{f}(\tilde{w}_{t-2}) - \eta\mathcal{H}\xi_{t-1} - \eta\mathcal{H}(1 - \eta\mathcal{H})\xi_{t-2} = \cdots \\
&= (1 - \eta\mathcal{H})^t \nabla f(w_0) - \eta\mathcal{H} \sum_{\tau=0}^{t-1} (1 - \eta\mathcal{H})^{t-\tau-1} \xi_\tau
\end{aligned} \tag{A.24}$$

Therefore, we have:

$$\begin{aligned}
\tilde{w}_t - w_0 &= -\eta \sum_{\tau=0}^{t-1} (\nabla \tilde{f}(\tilde{w}_\tau) + \xi_\tau) \\
&= -\eta \sum_{\tau=0}^{t-1} \left((1 - \eta\mathcal{H})^\tau \nabla f(w_0) - \eta\mathcal{H} \sum_{\tau'=0}^{\tau-1} (1 - \eta\mathcal{H})^{\tau-\tau'-1} \xi_{\tau'} + \xi_\tau \right) \\
&= -\eta \sum_{\tau=0}^{t-1} (1 - \eta\mathcal{H})^\tau \nabla f(w_0) - \eta \sum_{\tau=0}^{t-1} (1 - \eta\mathcal{H})^{t-\tau-1} \xi_\tau
\end{aligned} \tag{A.25}$$

Next, we prove the existence of T in Eq.(A.21). Since $\sum_{\tau=0}^{t}(1+\eta\gamma_0)^{2\tau}$ is monotonically increasing w.r.t t, and diverge to infinity as $t \to \infty$. We know there is always some $T \in \mathbb{N}$ gives $\frac{d}{\eta\gamma_0} \le \sum_{\tau=0}^{T-1}(1+\eta\gamma_0)^{2\tau}$. Let T be the smallest integer satisfying above equation. By assumption, we know $\gamma \le \gamma_0 \le L$, and

$$\sum_{\tau=0}^{t+1}(1+\eta\gamma_0)^{2\tau} = 1 + (1+\eta\gamma_0)^2 \sum_{\tau=0}^{t}(1+\eta\gamma_0)^{2\tau} \tag{A.26}$$

we can choose $\eta_{\max} < \min\{(\sqrt{2}-1)/L, 2d/\gamma\}$ so that

$$\frac{d}{\eta\gamma_0} \le \sum_{\tau=0}^{T-1}(1+\eta\gamma_0)^{2\tau} \le 1 + \frac{2d}{\eta\gamma_0} \le \frac{3d}{\eta\gamma_0} \tag{A.27}$$

Finally, by Eq.(A.21), we know $T = O(\log d/\gamma_0\eta)$, and $(1+\eta\gamma_0)^T \le \tilde{O}(1)$. Also because $\mathbb{E}\xi = 0$ and $\|\xi\| \le Q = \tilde{O}(1)$ with probability 1, then by Hoeffding inequality, we have for each dimension i and time $t \le T$:

$$P\left(|\eta \sum_{\tau=0}^{t-1}(1-\eta\mathcal{H})^{t-\tau-1}\xi_{\tau,i}| > \tilde{O}(\eta^{\frac{1}{2}}\log\frac{1}{\eta})\right) \le e^{-\tilde{\Omega}(\log^2\frac{1}{\eta})} \le \tilde{O}(\eta^4) \tag{A.28}$$

then by summing over dimension d and taking union bound over all $t \le T$, we directly have:

$$P\left(\forall t \le T, \|\eta \sum_{\tau=0}^{t-1}(1-\eta\mathcal{H})^{t-\tau-1}\xi_{\tau}\| > \tilde{O}(\eta^{\frac{1}{2}}\log\frac{1}{\eta})\right) \le \tilde{O}(\eta^3). \tag{A.29}$$

Combine this fact with Eq.(A.24) and Eq.(A.25), we finish the proof.

□

Next we need to prove that the two sequences of updates are always close.

Lemma A.1.6. *Under the assumptions of Theorem A.1.1. and let $\{w_t\}$ be the corresponding sequence generated by running SGD on function f. Also let $\tilde{f}$ and $\{\tilde{w}_t\}$ be defined as in Lemma A.1.5. Then, for any initial point w_0 where $\|\nabla f(w_0)\| \le \tilde{O}(\eta) < \varepsilon$, and $\lambda_{\min}(\nabla^2 f(w_0)) = -\gamma_0$.*

Given the choice of T as in Eq.(A.21), with probability at least $1 - \tilde{O}(\eta^2)$, we have following holds simultaneously for all $t \le T$:

$$\|w_t - \tilde{w}_t\| \le \tilde{O}(\eta \log^2 \frac{1}{\eta}); \qquad \|\nabla f(w_t) - \nabla \tilde{f}(\tilde{w}_t)\| \le \tilde{O}(\eta \log^2 \frac{1}{\eta}) \tag{A.30}$$

Proof. First, we have update function of gradient by:

$$\begin{aligned}\nabla f(w_t) =& \nabla f(w_{t-1}) + \int_0^1 \mathcal{H}(w_{t-1} + t(w_t - w_{t-1}))\mathrm{d}t \cdot (w_t - w_{t-1}) \\ =& \nabla f(w_{t-1}) + \mathcal{H}(w_{t-1})(w_t - w_{t-1}) + \theta_{t-1}\end{aligned} \tag{A.31}$$

where the remainder:

$$\theta_{t-1} \equiv \int_0^1 \left[\mathcal{H}(w_{t-1} + t(w_t - w_{t-1})) - \mathcal{H}(w_{t-1})\right] \mathrm{d}t \cdot (w_t - w_{t-1}) \tag{A.32}$$

Denote $\mathcal{H} = \mathcal{H}(w_0)$, and $\mathcal{H}'_{t-1} = \mathcal{H}(w_{t-1}) - \mathcal{H}(w_0)$. By Hessian smoothness, we immediately have:

$$\|\mathcal{H}'_{t-1}\| = \|\mathcal{H}(w_{t-1}) - \mathcal{H}(w_0)\| \le \rho\|w_{t-1} - w_0\| \le \rho(\|w_t - \tilde{w}_t\| + \|\tilde{w}_t - w_0\|) \tag{A.33}$$

$$\|\theta_{t-1}\| \le \frac{\rho}{2}\|w_t - w_{t-1}\|^2 \tag{A.34}$$

Substitute the update equation of SGD (Eq.(A.1)) into Eq.(A.31), we have:

$$\begin{aligned}\nabla f(w_t) &= \nabla f(w_{t-1}) - \eta(\mathcal{H} + \mathcal{H}'_{t-1})(\nabla f(w_{t-1}) + \xi_{t-1}) + \theta_{t-1} \\ &= (1 - \eta\mathcal{H})\nabla f(w_{t-1}) - \eta\mathcal{H}\xi_{t-1} - \eta\mathcal{H}'_{t-1}(\nabla f(w_{t-1}) + \xi_{t-1}) + \theta_{t-1}\end{aligned} \tag{A.35}$$

Let $\Delta_t = \nabla f(w_t) - \nabla \tilde{f}(\tilde{w}_t)$ denote the difference in gradient, then from Eq.(A.24), Eq.(A.35), and Eq.(A.1), we have:

$$\Delta_t = (1 - \eta\mathcal{H})\Delta_{t-1} - \eta\mathcal{H}'_{t-1}[\Delta_{t-1} + \nabla \tilde{f}(\tilde{w}_{t-1}) + \xi_{t-1}] + \theta_{t-1} \tag{A.36}$$

$$w_t - \tilde{w}_t = -\eta \sum_{\tau=0}^{t-1} \Delta_\tau \tag{A.37}$$

Let filtration $\mathfrak{F}_t = \sigma\{\xi_0, \cdots \xi_{t-1}\}$, and note $\sigma\{\Delta_0, \cdots, \Delta_t\} \subset \mathfrak{F}_t$, where $\sigma\{\cdot\}$ denotes the sigma field. Also, let event $\mathfrak{K}_t = \{\forall \tau \leq t,\ \|\nabla \tilde{f}(\tilde{w}_\tau)\| \leq \tilde{O}(\eta^{\frac{1}{2}} \log \frac{1}{\eta}),\ \|\tilde{w}_\tau - w_0\| \leq \tilde{O}(\eta^{\frac{1}{2}} \log \frac{1}{\eta})\}$, and $\mathfrak{E}_t = \{\forall \tau \leq t,\ \|\Delta_\tau\| \leq \mu\eta \log^2 \frac{1}{\eta}\}$, where μ is independent of (η, ζ), and will be specified later. Again, $\tilde{O}$ notation in this proof will never hide any dependence on μ. Clearly, we have $\mathfrak{K}_t \subset \mathfrak{K}_{t-1}$ ($\mathfrak{E}_t \subset \mathfrak{E}_{t-1}$), thus $1_{\mathfrak{K}_t} \leq 1_{\mathfrak{K}_{t-1}}$ ($1_{\mathfrak{E}_t} \leq 1_{\mathfrak{E}_{t-1}}$), where $1_{\mathfrak{K}}$ is the indicator function of event $\mathfrak{K}$.

We first need to carefully bounded all terms in Eq.(A.36), conditioned on event $\mathfrak{K}_{t-1} \cap \mathfrak{E}_{t-1}$, by Eq.(A.33), Eq.(A.34)), and Eq.(A.37), with probability 1, for all $t \leq T \leq O(\log d/\gamma_0\eta)$, we have:

$$\begin{aligned} &\|(1 - \eta\mathcal{H})\Delta_{t-1}\| \leq \tilde{O}(\mu\eta \log^2 \frac{1}{\eta}) \qquad &&\|\eta\mathcal{H}'_{t-1}(\Delta_{t-1} + \nabla \tilde{f}(\tilde{w}_{t-1}))\| \leq \tilde{O}(\eta^2 \log^2 \frac{1}{\eta}) \\ &\|\eta\mathcal{H}'_{t-1}\xi_{t-1}\| \leq \tilde{O}(\eta^{1.5} \log \frac{1}{\eta}) \qquad &&\|\theta_{t-1}\| \leq \tilde{O}(\eta^2) \end{aligned} \tag{A.38}$$

Since event $\mathfrak{K}_{t-1} \subset \mathfrak{F}_{t-1}, \mathfrak{E}_{t-1} \subset \mathfrak{F}_{t-1}$ thus independent of ξ_{t-1}, we also have:

$$\begin{aligned} &\mathbb{E}[((1 - \eta\mathcal{H})\Delta_{t-1})^T \eta\mathcal{H}'_{t-1}\xi_{t-1} 1_{\mathfrak{K}_{t-1} \cap \mathfrak{E}_{t-1}} \mid \mathfrak{F}_{t-1}] \\ =&1_{\mathfrak{K}_{t-1} \cap \mathfrak{E}_{t-1}}((1 - \eta\mathcal{H})\Delta_{t-1})^T \eta\mathcal{H}'_{t-1}\mathbb{E}[\xi_{t-1} \mid \mathfrak{F}_{t-1}] = 0 \end{aligned} \tag{A.39}$$

Therefore, from Eq.(A.36) and Eq.(A.38):

$$\begin{aligned} &\mathbb{E}[\|\Delta_t\|_2^2 1_{\mathfrak{K}_{t-1} \cap \mathfrak{E}_{t-1}} \mid \mathfrak{F}_{t-1}] \\ \leq &\left[(1 + \eta\gamma_0)^2\|\Delta_{t-1}\|^2 + (1 + \eta\gamma_0)\|\Delta_{t-1}\|\tilde{O}(\eta^2 \log^2 \frac{1}{\eta}) + \tilde{O}(\eta^3 \log^2 \frac{1}{\eta})\right] 1_{\mathfrak{K}_{t-1} \cap \mathfrak{E}_{t-1}} \\ \leq &\left[(1 + \eta\gamma_0)^2\|\Delta_{t-1}\|^2 + \tilde{O}(\mu\eta^3 \log^4 \frac{1}{\eta})\right] 1_{\mathfrak{K}_{t-1} \cap \mathfrak{E}_{t-1}} \end{aligned} \tag{A.40}$$

Define

$$G_t = (1 + \eta\gamma_0)^{-2t}[\ \|\Delta_t\|^2 + \lambda\eta^2 \log^4 \frac{1}{\eta}\ ] \tag{A.41}$$

Then, when $\eta_{\max}$ is small enough, we have:

$$\mathbb{E}[G_t 1_{\mathfrak{K}_{t-1} \cap \mathfrak{E}_{t-1}} \mid \mathfrak{F}_{t-1}] = (1 + \eta\gamma_0)^{-2t}\left[\mathbb{E}[\|\Delta_t\|_2^2 1_{\mathfrak{K}_{t-1} \cap \mathfrak{E}_{t-1}} \mid \mathfrak{F}_{t-1}] + \lambda\eta^2 \log^3 \frac{1}{\eta}\right] 1_{\mathfrak{K}_{t-1} \cap \mathfrak{E}_{t-1}}$$

$$
\begin{aligned}
\leq & (1+\eta\gamma_0)^{-2t}\left[(1+\eta\gamma_0)^2\|\Delta_{t-1}\|^2+\tilde{O}(\mu\eta^3\log^4\frac{1}{\eta})+\lambda\eta^2\log^4\frac{1}{\eta}\right]1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} \\
\leq & (1+\eta\gamma_0)^{-2t}\left[(1+\eta\gamma_0)^2\|\Delta_{t-1}\|^2+(1+\eta\gamma_0)^2\lambda\eta^2\log^4\frac{1}{\eta}\right]1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} \\
= & G_{t-1}1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} \leq G_{t-1}1_{\mathfrak{K}_{t-2}\cap\mathfrak{E}_{t-2}}
\end{aligned} \tag{A.42}
$$

Therefore, we have $\mathbb{E}[G_t 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} \mid \mathfrak{F}_{t-1}] \leq G_{t-1}1_{\mathfrak{K}_{t-2}\cap\mathfrak{E}_{t-2}}$ which means $G_t 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}}$ is a supermartingale.

On the other hand, we have:

$$
\Delta_t = (1-\eta H)\Delta_{t-1} - \eta\mathcal{H}'_{t-1}(\Delta_{t-1}+\nabla\tilde{f}(\tilde{w}_{t-1})) - \eta\mathcal{H}'_{t-1}\xi_{t-1} + \theta_{t-1} \tag{A.43}
$$

Once conditional on filtration $\mathfrak{F}_{t-1}$, the first two terms are deterministic, and only the third and fourth term are random. Therefore, we know, with probability 1:

$$
|\,\|\Delta_t\|_2^2 - \mathbb{E}[\|\Delta_t\|_2^2|\mathfrak{F}_{t-1}]\,|1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} \leq \tilde{O}(\mu\eta^{2.5}\log^3\frac{1}{\eta}) \tag{A.44}
$$

Where the main contribution comes from the product of the first term and third term. Then, with probability 1, we have:

$$
\begin{aligned}
& |G_t 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} - \mathbb{E}[G_t 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} \mid \mathfrak{F}_{t-1}]| \\
= & (1+2\eta\gamma_0)^{-2t}\cdot|\,\|\Delta_t\|_2^2 - \mathbb{E}[\|\Delta_t\|_2^2|\mathfrak{F}_{t-1}]\,|\cdot 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} \leq \tilde{O}(\mu\eta^{2.5}\log^3\frac{1}{\eta}) = c_{t-1}
\end{aligned} \tag{A.45}
$$

By Azuma-Hoeffding inequality, with probability less than $\tilde{O}(\eta^3)$, for $t \leq T \leq O(\log d/\gamma_0\eta)$:

$$
G_t 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} - G_0\cdot 1 > \tilde{O}(1)\sqrt{\sum_{\tau=0}^{t-1}c_\tau^2\log(\frac{1}{\eta})} = \tilde{O}(\mu\eta^2\log^4\frac{1}{\eta}) \tag{A.46}
$$

This means there exist some $\tilde{C} = \tilde{O}(1)$ so that:

$$
P\left(G_t 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} \geq \tilde{C}\mu\eta^2\log^4\frac{1}{\eta}\right) \leq \tilde{O}(\eta^3) \tag{A.47}
$$

By choosing $\mu > \tilde{C}$, this is equivalent to:

$$
P\left(\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}\cap\left\{\|\Delta_t\|^2 \geq \mu^2\eta^2\log^4\frac{1}{\eta}\right\}\right) \leq \tilde{O}(\eta^3) \tag{A.48}
$$

Therefore, combined with Lemma A.1.5, we have:

$$\begin{aligned}
&P\left(\mathfrak{E}_{t-1} \cap \left\{\|\Delta_t\| \geq \mu\eta \log^2 \frac{1}{\eta}\right\}\right) \\
=&P\left(\mathfrak{K}_{t-1} \cap \mathfrak{E}_{t-1} \cap \left\{\|\Delta_t\| \geq \mu\eta \log^2 \frac{1}{\eta}\right\}\right) + P\left(\overline{\mathfrak{K}}_{t-1} \cap \mathfrak{E}_{t-1} \cap \left\{\|\Delta_t\| \geq \mu\eta \log^2 \frac{1}{\eta}\right\}\right) \\
\leq&\tilde{O}(\eta^3) + P(\overline{\mathfrak{K}}_{t-1}) \leq \tilde{O}(\eta^3)
\end{aligned} \tag{A.49}$$

Finally, we know:

$$P(\overline{\mathfrak{E}}_t) = P\left(\mathfrak{E}_{t-1} \cap \left\{\|\Delta_t\| \geq \mu\eta \log^2 \frac{1}{\eta}\right\}\right) + P(\overline{\mathfrak{E}}_{t-1}) \leq \tilde{O}(\eta^3) + P(\overline{\mathfrak{E}}_{t-1}) \tag{A.50}$$

Because $P(\overline{\mathfrak{E}}_0) = 0$, and $T \leq \tilde{O}(\frac{1}{\eta})$, we have $P(\overline{\mathfrak{E}}_T) \leq \tilde{O}(\eta^2)$. Due to Eq.(A.37), we have $\|w_t - \tilde{w}_t\| \leq \eta \sum_{\tau=0}^{t-1} \|\Delta_\tau\|$, then by the definition of $\mathfrak{E}_T$, we finish the proof.

□

Using the two lemmas above we are ready to prove Lemma A.1.4

Proof of Lemma A.1.4. Let $\tilde{f}$ and $\{\tilde{w}_t\}$ be defined as in Lemma A.1.5. and also let $\lambda_{\min}(\mathcal{H}(w_0)) = -\gamma_0$. Since $\mathcal{H}(w)$ is ρ-Lipschitz, for any w, w_0, we have:

$$f(w) \leq f(w_0) + \nabla f(w_0)^T(w - w_0) + \frac{1}{2}(w - w_0)^T\mathcal{H}(w_0)(w - w_0) + \frac{\rho}{6}\|w - w_0\|^3 \tag{A.51}$$

Denote $\tilde{\delta} = \tilde{w}_T - w_0$ and $\delta = w_T - \tilde{w}_T$, we have:

$$\begin{aligned}
f(w_T) - f(w_0) \leq& \left[\nabla f(w_0)^T(w_T - w_0) + \frac{1}{2}(w_T - w_0)^T\mathcal{H}(w_0)(w_T - w_0) + \frac{\rho}{6}\|w_T - w_0\|^3\right] \\
=& \left[\nabla f(w_0)^T(\tilde{\delta} + \delta) + \frac{1}{2}(\tilde{\delta} + \delta)^T\mathcal{H}(\tilde{\delta} + \delta) + \frac{\rho}{6}\|\tilde{\delta} + \delta\|^3\right] \\
=& \left[\nabla f(w_0)^T\tilde{\delta} + \frac{1}{2}\tilde{\delta}^T\mathcal{H}\tilde{\delta}\right] + \left[\nabla f(w_0)^T\delta + \tilde{\delta}^T\mathcal{H}\delta + \frac{1}{2}\delta^T\mathcal{H}\delta + \frac{\rho}{6}\|\tilde{\delta} + \delta\|^3\right]
\end{aligned} \tag{A.52}$$

Where $\mathcal{H} = \mathcal{H}(w_0)$. Denote $\tilde{\Lambda} = \nabla f(w_0)^T\tilde{\delta} + \frac{1}{2}\tilde{\delta}^T\mathcal{H}\tilde{\delta}$ be the first term, and $\Lambda = \nabla f(w_0)^T\delta + \tilde{\delta}^T\mathcal{H}\delta + \frac{1}{2}\delta^T\mathcal{H}\delta + \frac{\rho}{6}\|\tilde{\delta} + \delta\|^3$ be the second term. We have $f(w_T) - f(w_0) \leq \tilde{\Lambda} + \Lambda$.

Let $\mathfrak{E}_t = \{\forall \tau \le t, \|\tilde{w}_\tau - w_0\| \le \tilde{O}(\eta^{\frac{1}{2}} \log \frac{1}{\eta}),\ \|w_t - \tilde{w}_t\| \le \tilde{O}(\eta \log^2 \frac{1}{\eta})\}$, by the result of Lemma A.1.5 and Lemma A.1.6, we know $P(\mathfrak{E}_T) \ge 1 - \tilde{O}(\eta^2)$. Then, clearly, we have:

$$\begin{aligned}
\mathbb{E}f(w_T) - f(w_0) =& \mathbb{E}[f(w_T) - f(w_0)]1_{\mathfrak{E}_T} + \mathbb{E}[f(w_T) - f(w_0)]1_{\overline{\mathfrak{E}}_T} \\
\le& \mathbb{E}\tilde{\Lambda}1_{\mathfrak{E}_T} + \mathbb{E}\Lambda 1_{\mathfrak{E}_T} + \mathbb{E}[f(w_T) - f(w_0)]1_{\overline{\mathfrak{E}}_T} \\
=& \mathbb{E}\tilde{\Lambda} + \mathbb{E}\Lambda 1_{\mathfrak{E}_T} + \mathbb{E}[f(w_T) - f(w_0)]1_{\overline{\mathfrak{E}}_T} - \mathbb{E}\tilde{\Lambda}1_{\overline{\mathfrak{E}}_T}
\end{aligned} \tag{A.53}$$

We will carefully caculate $\mathbb{E}\tilde{\Lambda}$ term first, and then bound remaining term as "perturbation" to first term.

Let $\lambda_1, \cdots, \lambda_d$ be the eigenvalues of $\mathcal{H}$. By the result of lemma A.1.5 and simple linear algebra, we have:

$$\begin{aligned}
\mathbb{E}\tilde{\Lambda} &= -\frac{\eta}{2}\sum_{i=1}^{d}\sum_{\tau=0}^{2T-1}(1-\eta\lambda_i)^\tau |\nabla_i f(w_0)|^2 + \frac{1}{2}\sum_{i=1}^{d}\lambda_i \sum_{\tau=0}^{T-1}(1-\eta\lambda_i)^{2\tau}\eta^2\sigma^2 \\
&\le \frac{1}{2}\sum_{i=1}^{d}\lambda_i \sum_{\tau=0}^{T-1}(1-\eta\lambda_i)^{2\tau}\eta^2\sigma^2 \\
&\le \frac{\eta^2\sigma^2}{2}\left[\frac{d-1}{\eta} - \gamma_0\sum_{\tau=0}^{T-1}(1+\eta\gamma_0)^{2\tau}\right] \le -\frac{\eta\sigma^2}{2}
\end{aligned} \tag{A.54}$$

The last inequality is directly implied by the choice of T as in Eq.(A.21). Also, by Eq.(A.21), we also immediately have that $T = O(\log d/\gamma_0\eta) \le O(\log d/\gamma\eta)$. Therefore, by choose $T_{max} = O(\log d/\gamma\eta)$ with large enough constant, we have $T \le T_{max} = O(\log d/\gamma\eta)$.

For bounding the second term, by definition of $\mathfrak{E}_t$, we have:

$$\mathbb{E}\Lambda 1_{\mathfrak{E}_T} = \mathbb{E}\left[\nabla f(w_0)^T\delta + \tilde{\delta}^T\mathcal{H}\delta + \frac{1}{2}\delta^T\mathcal{H}\delta + \frac{\rho}{6}\|\tilde{\delta}+\delta\|^3\right]1_{\mathfrak{E}_T} \le \tilde{O}(\eta^{1.5}\log^3\frac{1}{\eta}) \tag{A.55}$$

On the other hand, since noise is bounded as $\|\xi\| \le \tilde{O}(1)$, from the results of Lemma A.1.5, it's easy to show $\|\tilde{w} - w_0\| = \|\tilde{\delta}\| \le \tilde{O}(1)$ is also bounded with probability 1. Recall the assumption that

function f is also bounded, then we have:

$$\begin{aligned}
&\mathbb{E}[f(w_T) - f(w_0)]1_{\overline{\mathfrak{E}}_T} - \mathbb{E}\tilde{\Lambda}1_{\overline{\mathfrak{E}}_T} \\
=&\mathbb{E}[f(w_T) - f(w_0)]1_{\overline{\mathfrak{E}}_T} - \mathbb{E}\left[\nabla f(w_0)^T\tilde{\delta} + \frac{1}{2}\tilde{\delta}^T\mathcal{H}\tilde{\delta}\right]1_{\overline{\mathfrak{E}}_T} \leq \tilde{O}(1)P(\overline{\mathfrak{E}}_T) \leq \tilde{O}(\eta^2)
\end{aligned} \tag{A.56}$$

Finally, substitute Eq.(A.54), Eq.(A.55) and Eq.(A.56) into Eq.(A.53), we finish the proof. □

Finally, we combine three cases to prove the main theorem.

Proof of Theorem A.1.1. Let's set $\mathcal{L}_1 = \{w \mid \|\nabla f(w)\| \geq \sqrt{2\eta\sigma^2 Ld}\}$, $\mathcal{L}_2 = \{w \mid \|\nabla f(w)\| \leq \sqrt{2\eta\sigma^2 Ld}$ and $\lambda_{\min}(\mathcal{H}(w)) \leq -\gamma\}$, and $\mathcal{L}_3 = \mathcal{L}_1^c \cup \mathcal{L}_2^c$. By choosing small enough $\eta_{\max}$, we could make $\sqrt{2\eta\sigma^2 Ld} < \min\{\varepsilon, \lambda\delta\}$. Under this choice, we know from Definition 3.2.2 of $(\lambda, \gamma, \varepsilon, \delta)$-strict saddlethat $\mathcal{L}_3$ is the locally λ-strongly convex region which is $\tilde{O}(\sqrt{\eta})$-close to some local minimum.

We shall first prove that within $\tilde{O}(\frac{1}{\eta^2}\log\frac{1}{\zeta})$ steps with probability at least $1 - \zeta/2$ one of w_t is in $\mathcal{L}_3$. Then by Lemma A.1.3 we know with probability at most $\zeta/2$ there exists a w_t that is in $\mathcal{L}_3$ but the last point is not. By union bound we will get the main result.

To prove within $\tilde{O}(\frac{1}{\eta^2}\log\frac{1}{\zeta})$ steps with probability at least $1 - \zeta/2$ one of w_t is in $\mathcal{L}_3$, we first show starting from any point, in $\tilde{O}(\frac{1}{\eta^2})$ steps with probability at least $1/2$ one of w_t is in $\mathcal{L}_3$. Then we can repeat this $\log 1/\zeta$ times to get the high probability result.

Define stochastic process $\{\tau_i\}$ s.t. $\tau_0 = 0$, and

$$\tau_{i+1} = \begin{cases} \tau_i + 1 & \text{if } w_{\tau_i} \in \mathcal{L}_1 \cup \mathcal{L}_3 \\ \tau_i + T(w_{\tau_i}) & \text{if } w_{\tau_i} \in \mathcal{L}_2 \end{cases} \tag{A.57}$$

Where $T(w_{\tau_i})$ is defined by Eq.(A.21) with $\gamma_0 = \lambda_{\min}(\mathcal{H}(w_{\tau_i}))$and we know $T \leq T_{max} = \tilde{O}(\frac{1}{\eta})$.

By Lemma A.1.2 and Lemma A.1.4, we know:

$$\mathbb{E}[f(w_{\tau_{i+1}}) - f(w_{\tau_i})|w_{\tau_i} \in \mathcal{L}_1, \mathfrak{F}_{\tau_i - 1}] = \mathbb{E}[f(w_{\tau_{i+1}}) - f(w_{\tau_i})|w_{\tau_i} \in \mathcal{L}_1] \leq -\tilde{O}(\eta^2) \tag{A.58}$$

$$\mathbb{E}[f(w_{\tau_{i+1}}) - f(w_{\tau_i})|w_{\tau_i} \in \mathcal{L}_2, \mathfrak{F}_{\tau_i - 1}] = \mathbb{E}[f(w_{\tau_{i+1}}) - f(w_{\tau_i})|w_{\tau_i} \in \mathcal{L}_2] \leq -\tilde{O}(\eta) \tag{A.59}$$

Therefore, combine above equation, we have:

$$\mathbb{E}[f(w_{\tau_{i+1}}) - f(w_{\tau_i})|w_{\tau_i} \notin \mathcal{L}_3, \mathfrak{F}_{\tau_i - 1}] = \mathbb{E}[f(w_{\tau_{i+1}}) - f(w_{\tau_i})|w_{\tau_i} \notin \mathcal{L}_3] \leq -(\tau_{i+1} - \tau_i)\tilde{O}(\eta^2) \tag{A.60}$$

Define event $\mathfrak{E}_i = \{\exists j \leq i,\ w_{\tau_j} \in \mathcal{L}_3\}$, clearly $\mathfrak{E}_i \subset \mathfrak{E}_{i+1}$, thus $P(\mathfrak{E}_i) \leq P(\mathfrak{E}_{i+1})$. Finally, consider $f(w_{\tau_{i+1}})1_{\mathfrak{E}_i}$, we have:

$$\begin{aligned}\mathbb{E}f(w_{\tau_{i+1}})1_{\mathfrak{E}_i} - \mathbb{E}f(w_{\tau_i})1_{\mathfrak{E}_{i-1}} &\leq B \cdot P(\mathfrak{E}_i - \mathfrak{E}_{i-1}) + \mathbb{E}[f(w_{\tau_{i+1}}) - f(w_{\tau_i})|\overline{\mathfrak{E}_i}] \cdot P(\overline{\mathfrak{E}_i}) \\ &\leq B \cdot P(\mathfrak{E}_i - \mathfrak{E}_{i-1}) - (\tau_{i+1} - \tau_i)\tilde{O}(\eta^2)P(\overline{\mathfrak{E}_i}) \end{aligned} \tag{A.61}$$

Therefore, by summing up over i, we have:

$$\mathbb{E}f(w_{\tau_i})1_{\mathfrak{E}_i} - f(w_0) \leq BP(\mathfrak{E}_i) - \tau_i\tilde{O}(\eta^2)P(\overline{\mathfrak{E}_i}) \leq B - \tau_i\tilde{O}(\eta^2)P(\overline{\mathfrak{E}_i}) \tag{A.62}$$

Since $|f(w_{\tau_i})1_{\mathfrak{E}_i}| < B$ is bounded, as τ_i grows to as large as $\frac{6B}{\eta^2}$, we must have $P(\overline{\mathfrak{E}_i}) < \frac{1}{2}$. That is, after $\tilde{O}(\frac{1}{\eta^2})$ steps, with at least probability $1/2$, $\{w_t\}$ have at least enter $\mathcal{L}_3$ once. Since this argument holds for any starting point, we can repeat this $\log 1/\zeta$ times and we know after $\tilde{O}(\frac{1}{\eta^2}\log 1/\zeta)$ steps, with probability at least $1 - \zeta/2$, $\{w_t\}$ have at least enter $\mathcal{L}_3$ once.

Combining with Lemma A.1.3, and by union bound we know after $\tilde{O}(\frac{1}{\eta^2}\log 1/\zeta)$ steps, with probability at least $1 - \zeta$, w_t will be in the $\tilde{O}(\sqrt{\eta \log \frac{1}{\eta\zeta}})$ neigborhood of some local minimum. □

A.2 Detailed analysis for Section 3.2 in constrained case

So far, we have been discussed all about unconstrained problem. In this section we extend our result to equality constraint problems under some mild conditions.

Consider the equality constrained optimization problem:

$$\begin{aligned} \min_{w} \quad & f(w) \\ \text{s.t.} \quad & c_i(w) = 0, \qquad i = 1, \cdots, m \end{aligned} \tag{A.63}$$

Define the feasible set as the set of points that satisfy all the constraints $\mathcal{W} = \{w \mid c_i(w) = 0;\ i = 1, \cdots, m\}$.

In this case, the algorithm we are running is Projected Noisy Gradient Descent. Let function $\Pi_{\mathcal{W}}(v)$ to be the projection to the feasible set, where the projection is defined as the global solution of $\min_{w \in \mathcal{W}} \|v - w\|^2$.

With same argument as in the unconstrained case, we could slightly simplify and convert it to standard projected stochastic gradient descent (PSGD) with update equation:

$$v_t = w_{t-1} - \eta \nabla f(w_{t-1}) + \xi_{t-1} \tag{A.64}$$

$$w_t = \Pi_{\mathcal{W}}(v_t) \tag{A.65}$$

As in unconstrained case, we are interested in noise ξ is i.i.d satisfying $\mathbb{E}\xi = 0$, $\mathbb{E}\xi\xi^T = \sigma^2 I$ and $\|\xi\| \leq Q$ almost surely. Our proof can be easily extended to Algorithm 2 with $\frac{1}{d} I \preceq \mathbb{E}\xi\xi^T \preceq (Q + \frac{1}{d})I$. In this section we first introduce basic tools for handling constrained optimization problems (most these materials can be found in [127]), then we prove some technical lemmas that are useful for dealing with the projection step in PSGD, finally we point out how to modify the previous analysis.

A.2.1 Preliminaries

Often for constrained optimization problems we want the constraints to satisfy some regularity conditions. LICQ (linear independent constraint quantification) is a common assumption in this context.

Definition A.2.1 (LICQ)**.** *In equality-constraint problem Eq.(A.63), given a point w, we say that the linear independence constraint qualification (LICQ) holds if the set of constraint gradients* $\{\nabla c_i(x), i = 1, \cdots, m\}$ *is linearly independent.*

In constrained optimization, we can locally transform it to an unconstrained problem by introducing Lagrangian multipliers. The Langrangian $\mathcal{L}$ can be written as

$$\mathcal{L}(w, \lambda) = f(w) - \sum_{i=1}^{m} \lambda_i c_i(w) \tag{A.66}$$

Then, if LICQ holds for all $w \in \mathcal{W}$, we can properly define function $\lambda^*(\cdot)$ to be:

$$\lambda^*(w) = \arg\min_{\lambda} \|\nabla f(w) - \sum_{i=1}^{m} \lambda_i \nabla c_i(w)\| = \arg\min_{\lambda} \|\nabla_w \mathcal{L}(w, \lambda)\| \tag{A.67}$$

where $\lambda^*(\cdot)$ can be calculated analytically: let matrix $C(w) = (\nabla c_1(w), \cdots, \nabla c_m(w))$, then we have:

$$\lambda^*(w) = C(w)^{\dagger} \nabla f(w) = (C(w)^T C(w))^{-1} C(w)^T \nabla f(w) \tag{A.68}$$

where $(\cdot)^{\dagger}$ is Moore-Penrose pseudo-inverse.

In our setting we need a stronger regularity condition which we call robust LICQ (RLICQ).

Definition A.2.2 (α_c-RLICQ)**.** *In equality-constraint problem Eq.(A.63), given a point w, we say that* α_c*-robust linear independence constraint qualification (* α_c*-RLICQ) holds if the minimum singular value of matrix* $C(w) = (\nabla c_1(w), \cdots, \nabla c_m(w))$ *is greater or equal to* α_c*, that is* $\sigma_{\min}(C(w)) \geq \alpha_c$.

Remark 3. *Given a point $w \in \mathcal{W}$, α_c-RLICQ implies LICQ. While LICQ holds for all $w \in \mathcal{W}$ is a necessary condition for $\lambda^*(w)$ to be well-defined; it's easy to check that α_c-RLICQ holds for all $w \in \mathcal{W}$ is a necessary condition for $\lambda^*(w)$ to be bounded. Later, we will also see α_c-RLICQ combined with the smoothness of $\{c_i(w)\}_{i=1}^m$ guarantee the curvature of constraint manifold to be bounded everywhere.*

Note that we require this condition in order to provide a quantitative bound, without this assumption there can be cases that are exponentially close to a function that does not satisfy LICQ.

We can also write down the first-order and second-order partial derivative of Lagrangian $\mathcal{L}$ at point $(w, \lambda^*(w))$:

$$\chi(w) = \nabla_w \mathcal{L}(w, \lambda)|_{(w,\lambda^*(w))} = \nabla f(w) - \sum_{i=1}^{m} \lambda_i^*(w) \nabla c_i(w) \tag{A.69}$$

$$\mathfrak{M}(w) = \nabla_{ww}^2 \mathcal{L}(w, \lambda)|_{(w,\lambda^*(w))} = \nabla^2 f(w) - \sum_{i=1}^{m} \lambda_i^*(w) \nabla^2 c_i(w) \tag{A.70}$$

Definition A.2.3 (Tangent Space and Normal Space)**.** *Given a feasible point $w \in \mathcal{W}$, define its corresponding Tangent Space to be $\mathcal{T}(w) = \{v \mid \nabla c_i(w)^T v = 0;\ i = 1, \cdots, m\}$, and Normal Space to be $\mathcal{T}^c(w) = span\{\nabla c_1(w) \cdots, \nabla c_m(w)\}$*

If $w \in \mathcal{R}^d$, and we have m constraint satisfying α_c-RLICQ , the tangent space would be a linear subspace with dimension $d - m$; and the normal space would be a linear subspace with dimension m. We also know immediately that $\chi(w)$ defined in Eq.(A.69) has another interpretation: it's the component of gradient $\nabla f(w)$ in tangent space.

Also, it's easy to see the normal space $\mathcal{T}^c(w)$ is the orthogonal complement of $\mathcal{T}$. We can also define the projection matrix of any vector onto tangent space (or normal space) to be $P_{\mathcal{T}(w)}$ (or $P_{\mathcal{T}^c(w)}$). Then, clearly, both $P_{\mathcal{T}(w)}$ and $P_{\mathcal{T}^c(w)}$ are orthoprojector, thus symmetric. Also by

Pythagorean theorem, we have:

$$\|v\|^2 = \|P_{\mathcal{T}(w)}v\|^2 + \|P_{\mathcal{T}^c(w)}v\|^2, \qquad \forall v \in \mathbb{R}^d \tag{A.71}$$

Taylor Expansion Let $w, w_0 \in \mathcal{W}$, and fix $\lambda^* = \lambda^*(w_0)$ independent of w, assume $\nabla^2_{ww}\mathcal{L}(w, \lambda^*)$ is ρ_L-Lipschitz, that is $\|\nabla^2_{ww}\mathcal{L}(w_1, \lambda^*) - \nabla^2_{ww}\mathcal{L}(w_2, \lambda^*)\| \leq \rho_L\|w_1 - w_2\|$ By Taylor expansion, we have:

$$\begin{aligned}\mathcal{L}(w, \lambda^*) \leq & \mathcal{L}(w_0, \lambda^*) + \nabla_w\mathcal{L}(w_0, \lambda^*)^T(w - w_0) \\ & + \frac{1}{2}(w - w_0)^T\nabla^2_{ww}\mathcal{L}(w_0, \lambda^*)(w - w_0) + \frac{\rho_L}{6}\|w - w_0\|^3\end{aligned} \tag{A.72}$$

Since w, w_0 are feasible, we know: $\mathcal{L}(w, \lambda^*) = f(w)$ and $\mathcal{L}(w_0, \lambda^*) = f(w_0)$, this gives:

$$f(w) \leq f(w_0) + \chi(w_0)^T(w - w_0) + \frac{1}{2}(w - w_0)^T\mathfrak{M}(w_0)(w - w_0) + \frac{\rho_L}{6}\|w - w_0\|^3 \tag{A.73}$$

Derivative of $\chi(w)$ By taking derative of $\chi(w)$ again, we know the change of this tangent gradient can be characterized by:

$$\nabla\chi(w) = \mathcal{H} - \sum_{i=1}^m \lambda_i^*(w)\nabla^2 c_i(w) - \sum_{i=1}^m \nabla c_i(w)\nabla\lambda_i^*(w)^T \tag{A.74}$$

Denote

$$\mathfrak{N}(w) = -\sum_{i=1}^m \nabla c_i(w)\nabla\lambda_i^*(w)^T \tag{A.75}$$

We immediately know that $\nabla\chi(w) = \mathfrak{M}(w) + \mathfrak{N}(w)$.

Remark 4. *The additional term $\mathfrak{N}(w)$ is not necessary to be even symmetric in general. This is due to the fact that $\chi(w)$ may not be the gradient of any scalar function. However, $\mathfrak{N}(w)$ has an important property that is: for any vector $v \in \mathbb{R}^d$, $\mathfrak{N}(w)v \in \mathcal{T}^c(w)$.*

Finally, for completeness, we state here the first/second-order necessary (or sufficient) conditions for optimality. Please refer to [127] for the proof of those theorems.

Theorem A.2.4 (First-Order Necessary Conditions)**.** *In equality constraint problem Eq.(A.63), suppose that $w^\dagger$ is a local solution, and that the functions f and c_i are continuously differentiable, and that the LICQ holds at $w^\dagger$. Then there is a Lagrange multiplier vector $\lambda^\dagger$, such that:*

$$\nabla_w \mathcal{L}(w^\dagger, \lambda^\dagger) = 0 \tag{A.76}$$

$$c_i(w^\dagger) = 0, \qquad \text{for } i = 1, \cdots, m \tag{A.77}$$

These conditions are also usually referred as Karush-Kuhn-Tucker (KKT) conditions.

Theorem A.2.5 (Second-Order Necessary Conditions)**.** *In equality constraint problem Eq.(A.63), suppose that $w^\dagger$ is a local solution, and that the LICQ holds at $w^\dagger$. Let $\lambda^\dagger$ Lagrange multiplier vector for which the KKT conditions are satisfied. Then:*

$$v^T \nabla^2_{xx} \mathcal{L}(w^\dagger, \lambda^\dagger) v \geq 0 \qquad \text{for all } v \in \mathcal{T}(w^\dagger) \tag{A.78}$$

Theorem A.2.6 (Second-Order Sufficient Conditions)**.** *In equality constraint problem Eq.(A.63), suppose that for some feasible point $w^\dagger \in \mathbb{R}^d$, and there's Lagrange multiplier vector $\lambda^\dagger$ for which the KKT conditions are satisfied. Suppose also that:*

$$v^T \nabla^2_{xx} \mathcal{L}(w^\dagger, \lambda^\dagger) v > 0 \qquad \text{for all } v \in \mathcal{T}(w^\dagger), v \neq 0 \tag{A.79}$$

Then $w^\dagger$ is a strict local solution.

Remark 5. *By definition Eq.(A.68), we know immediately $\lambda^*(w^\dagger)$ is one of valid Lagrange multipliers $\lambda^\dagger$ for which the KKT conditions are satisfied. This means $\chi(w^\dagger) = \nabla_w \mathcal{L}(w^\dagger, \lambda^\dagger)$ and $\mathfrak{M}(w^\dagger) = \mathcal{L}(w^\dagger, \lambda^\dagger)$.*

Therefore, Theorem A.2.4, A.2.5, A.2.6 gives strong implication that $\chi(w)$ and $\mathfrak{M}(w)$ are the right thing to look at, which are in some sense equivalent to $\nabla f(w)$ and $\nabla^2 f(w)$ in unconstrained case.

A.2.2 Geometrical lemmas regarding constraint manifold

Since in equality constraint problem, at each step of PSGD, we are effectively considering the local manifold around feasible point w_{t-1}. In this section, we provide some technical lemmas relating to the geometry of constraint manifold in preparsion for the proof of main theorem in equality constraint case.

We first show if two points are close, then the projection in the normal space is much smaller than the projection in the tangent space.

Lemma A.2.7. *Suppose the constraints $\{c_i\}_{i=1}^m$ are L_i-smooth, and α_c-RLICQ holds for all $w \in \mathcal{W}$. Then, let $\sum_{i=1}^m \frac{L_i^2}{\alpha_c^2} = \frac{1}{R^2}$, for any $w, w_0 \in \mathcal{W}$, let $\mathcal{T}_0 = \mathcal{T}(w_0)$, then*

$$\|P_{\mathcal{T}_0^c}(w - w_0)\| \leq \frac{1}{2R}\|w - w_0\|^2 \tag{A.80}$$

Furthermore, if $\|w - w_0\| < R$ holds, we additionally have:

$$\|P_{\mathcal{T}_0^c}(w - w_0)\| \leq \frac{\|P_{\mathcal{T}_0}(w - w_0)\|^2}{R} \tag{A.81}$$

Proof. First, since for any vector $\hat{v} \in \mathcal{T}_0$, we have $\|C(w_0)^T\hat{v}\| = 0$, then by simple linear algebra, it's easy to show:

$$\begin{aligned}\|C(w_0)^T(w - w_0)\|^2 =& \|C(w_0)^T P_{\mathcal{T}_0^c}(w - w_0)\|^2 \geq \sigma_{\min}^2 \|P_{\mathcal{T}_0^c}(w - w_0)\|^2 \\ \geq& \alpha_c^2 \|P_{\mathcal{T}_0^c}(w - w_0)\|^2\end{aligned} \tag{A.82}$$

On the other hand, by L_i-smooth, we have:

$$|c_i(w) - c_i(w_0) - \nabla c_i(w_0)^T(w - w_0)| \leq \frac{L_i}{2}\|w - w_0\|^2 \tag{A.83}$$

Since w, w_0 are feasible points, we have $c_i(w) = c_i(w_0) = 0$, which gives:

$$\|C(w_0)^T(w - w_0)\|^2 = \sum_{i=1}^m (\nabla c_i(w_0)^T(w - w_0))^2 \leq \sum_{i=1}^m \frac{L_i^2}{4}\|w - w_0\|^4 \tag{A.84}$$

Combining Eq.(A.82) and Eq.(A.84), and the definition of R, we have:

$$\|P_{\mathcal{T}_0^c}(w - w_0)\|^2 \leq \frac{1}{4R^2}\|w - w_0\|^4 = \frac{1}{4R^2}(\|P_{\mathcal{T}_0^c}(w - w_0)\|^2 + \|P_{\mathcal{T}_0}(w - w_0)\|^2)^2 \tag{A.85}$$

Solving this second-order inequality gives two solution

$$\|P_{\mathcal{T}_0^c}(w - w_0)\| \leq \frac{\|P_{\mathcal{T}_0}(w - w_0)\|^2}{R} \quad \text{or} \quad \|P_{\mathcal{T}_0^c}(w - w_0)\| \geq R \tag{A.86}$$

By assumption, we know $\|w - w_0\| < R$ (so the second case cannot be true), which finishes the proof. □

Here, we see the $\sqrt{\sum_{i=1}^m \frac{L_i^2}{\alpha_c^2}} = \frac{1}{R}$ serves as a upper bound of the curvatures on the constraint manifold, and equivalently, R serves as a lower bound of the radius of curvature. α_c-RLICQ and smoothness guarantee that the curvature is bounded.

Next we show the normal/tangent space of nearby points are close.

Lemma A.2.8. *Suppose the constraints $\{c_i\}_{i=1}^m$ are L_i-smooth, and α_c-RLICQ holds for all $w \in \mathcal{W}$. Let $\sum_{i=1}^m \frac{L_i^2}{\alpha_c^2} = \frac{1}{R^2}$, for any $w, w_0 \in \mathcal{W}$, let $\mathcal{T}_0 = \mathcal{T}(w_0)$. Then for all $\hat{v} \in \mathcal{T}(w)$ so that $\|\hat{v}\| = 1$, we have*

$$\|P_{\mathcal{T}_0^c} \cdot \hat{v}\| \leq \frac{\|w - w_0\|}{R} \tag{A.87}$$

Proof. With similar calculation as Eq.(A.82), we immediately have:

$$\|P_{\mathcal{T}_0^c} \cdot \hat{v}\|^2 \leq \frac{\|C(w_0)^T\hat{v}\|^2}{\sigma_{\min}^2(C(w))} \leq \frac{\|C(w_0)^T\hat{v}\|^2}{\alpha_c^2} \tag{A.88}$$

Since $\hat{v} \in \mathcal{T}(w)$, we have $C(w)^T\hat{v} = 0$, combined with the fact that $\hat{v}$ is a unit vector, we have:

$$\begin{aligned}\|C(w_0)^T\hat{v}\|^2 =& \|[C(w_0) - C(w)]^T\hat{v}\|^2 = \sum_{i=1}^m([\nabla c_i(w_0) - \nabla c_i(w)]^T\hat{v})^2 \\ &\leq \sum_{i=1}^m \|\nabla c_i(w_0) - \nabla c_i(w)\|^2\|\hat{v}\|^2 \leq \sum_{i=1}^m L_i^2\|w_0 - w\|^2\end{aligned} \tag{A.89}$$

Combining Eq.(A.88) and Eq.(A.89), and the definition of R, we concludes the proof. □

Lemma A.2.9. *Suppose the constraints $\{c_i\}_{i=1}^m$ are L_i-smooth, and α_c-RLICQ holds for all $w \in \mathcal{W}$. Let $\sum_{i=1}^m \frac{L_i^2}{\alpha_c^2} = \frac{1}{R^2}$, for any $w, w_0 \in \mathcal{W}$, let $\mathcal{T}_0 = \mathcal{T}(w_0)$. Then for all $\hat{v} \in \mathcal{T}^c(w)$ so that $\|\hat{v}\| = 1$, we have*

$$\|P_{\mathcal{T}_0} \cdot \hat{v}\| \leq \frac{\|w - w_0\|}{R} \tag{A.90}$$

Proof. By definition of projection, clearly, we have $P_{\mathcal{T}_0} \cdot \hat{v} + P_{\mathcal{T}_0^c} \cdot \hat{v} = \hat{v}$. Since $\hat{v} \in \mathcal{T}^c(w)$, without loss of generality, assume $\hat{v} = \sum_{i=1}^m \lambda_i \nabla c_i(w)$. Define $\tilde{d} = \sum_{i=1}^m \lambda_i \nabla c_i(w_0)$, clearly $\tilde{d} \in \mathcal{T}_0^c$. Since projection gives the closest point in subspace, we have:

$$\begin{aligned} \|P_{\mathcal{T}_0} \cdot \hat{v}\| =& \|P_{\mathcal{T}_0^c} \cdot \hat{v} - \hat{v}\| \leq \|\tilde{d} - \hat{v}\| \\ &\leq \sum_{i=1}^m \lambda_i \|\nabla c_i(w_0) - \nabla c_i(w)\| \leq \sum_{i=1}^m \lambda_i L_i \|w_0 - w\| \end{aligned} \tag{A.91}$$

On the other hand, let $\lambda = (\lambda_1, \cdots, \lambda_m)^T$, we know $C(w)\lambda = \hat{v}$, thus:

$$\lambda = C(w)^\dagger \hat{v} = (C(w)^T C(w))^{-1} C(w)^T \hat{v} \tag{A.92}$$

Therefore, by α_c-RLICQ and the fact $\hat{v}$ is unit vector, we know: $\|\lambda\| \leq \frac{1}{\alpha_c}$. Combined with Eq.(A.91), we finished the proof. □

Using the previous lemmas, we can then prove that: starting from any point w_0 on constraint manifold, the result of adding any small vector v and then projected back to feasible set, is not very different from the result of adding $P_{\mathcal{T}(w_0)} v$.

Lemma A.2.10. *Suppose the constraints $\{c_i\}_{i=1}^m$ are L_i-smooth, and α_c-RLICQ holds for all $w \in \mathcal{W}$. Let $\sum_{i=1}^m \frac{L_i^2}{\alpha_c^2} = \frac{1}{R^2}$, for any $w_0 \in \mathcal{W}$, let $\mathcal{T}_0 = \mathcal{T}(w_0)$. Then let $w_1 = w_0 + \eta\hat{v}$, and $w_2 = w_0 + \eta P_{\mathcal{T}_0} \cdot \hat{v}$, where $\hat{v} \in \mathbb{S}^{d-1}$ is a unit vector. Then, we have:*

$$\|\Pi_{\mathcal{W}}(w_1) - w_2\| \leq \frac{4\eta^2}{R} \tag{A.93}$$

Where projection $\Pi_{\mathcal{W}}(w)$ is defined as the closet point to w on feasible set $\mathcal{W}$.

Proof. First, note that $\|w_1 - w_0\| = \eta$, and by definition of projection, there must exist a project $\Pi_{\mathcal{W}}(w)$ inside the ball $\mathbb{B}_\eta(w_1) = \{w \mid \|w - w_1\| \leq \eta\}$.

Denote $u_1 = \Pi_{\mathcal{W}}(w_1)$, and clearly $u_1 \in \mathcal{W}$. we can formulate u_1 as the solution to following constrained optimization problems:

$$\begin{aligned} \min_{u} \quad & \|w_1 - u\|^2 \\ \text{s.t.} \quad & c_i(u) = 0, \qquad i = 1, \cdots, m \end{aligned} \tag{A.94}$$

Since function $f(u) = \|w_1 - u\|^2$ and $c_i(u)$ are continuously differentiable by assumption, and the condition α_c-RLICQ holds for all $w \in \mathcal{W}$ implies that LICQ holds for u_1. Therefore, by Karush-Kuhn-Tucker necessary conditions, we immediately know $(w_1 - u_1) \in \mathcal{T}^c(u_1)$.

Since $u_1 \in \mathbb{B}_\eta(w_1)$, we know $\|w_0 - u_1\| \leq 2\eta$, by Lemma A.2.9, we immediately have:

$$\|P_{\mathcal{T}_0}(w_1 - u_1)\| = \frac{\|P_{\mathcal{T}_0}(w_1 - u_1)\|}{\|w_1 - u_1\|}\|w_1 - u_1\| \leq \frac{1}{R}\|w_0 - u_1\| \cdot \|w_1 - u_1\| \leq \frac{2}{R}\eta^2 \tag{A.95}$$

Let $v_1 = w_0 + P_{\mathcal{T}_0}(u_1 - w_0)$, we have:

$$\begin{aligned} \|v_1 - w_2\| =& \|(v_1 - w_0) - (w_2 - w_0)\| = \|P_{\mathcal{T}_0}(u_1 - w_0) - P_{\mathcal{T}_0}(w_1 - w_0)\| \\ =& \|P_{\mathcal{T}_0}(w_1 - u_1)\| \leq \frac{2}{R}\eta^2 \end{aligned} \tag{A.96}$$

On the other hand by Lemma A.2.7, we have:

$$\|u_1 - v_1\| = \|P_{\mathcal{T}_0^c}(u_1 - w_0)\| \leq \frac{1}{2R}\|u_1 - w_0\|^2 \leq \frac{2}{R}\eta^2 \tag{A.97}$$

Combining Eq.(A.96) and Eq.(A.97), we finished the proof.

□

A.2.3 Main theorem

Now we are ready to prove the main theorems. First we revise the definition of strict saddle in the constrained case.

Definition A.2.11. *A twice differentiable function $f(w)$ with constraints $c_i(w)$ is* $(\lambda, \gamma, \varepsilon, \delta)$-strict saddle*, if for any point w one of the following is true*

1. $\|\chi(w)\| \geq \varepsilon$.
2. $\hat{v}^T \mathfrak{M}(w)\hat{v} \leq -\gamma$ *for some* $\hat{v} \in \mathcal{T}(w)$, $\|\hat{v}\| = 1$
3. *There is a local minimum $w^\star$ such that $\|w - w^\star\| \leq \delta$, and for all w' in the 2δ neighborhood of $w^\star$, we have $\hat{v}^T \mathfrak{M}(w')\hat{v} \geq \lambda$ for all $\hat{v} \in \mathcal{T}(w')$, $\|\hat{v}\| = 1$*

Next, we prove a equivalent formulation for PSGD.

Lemma A.2.12. *Suppose the constraints $\{c_i\}_{i=1}^m$ are L_i-smooth, and α_c-RLICQ holds for all $w \in \mathcal{W}$. Furthermore, if function f is L-Lipschitz, and the noise ξ is bounded, then running PSGD as in Eq.(A.64) is equivalent to running:*

$$w_t = w_{t-1} - \eta \cdot (\chi(w_{t-1}) + P_{\mathcal{T}(w_{t-1})}\xi_{t-1}) + \iota_{t-1} \tag{A.98}$$

where ι is the correction for projection, and $\|\iota\| \leq \tilde{O}(\eta^2)$.

Proof. Lemma A.2.12 is a direct corollary of Lemma A.2.10. □

The intuition behind this lemma is that: when $\{c_i\}_{i=1}^m$ are smooth and α_c-RLICQ holds for all $w \in \mathcal{W}$, then the constraint manifold has bounded curvature every where. Then, if we only care about first order behavior, it's well-approximated by the local dynamic in tangent plane, up to some second-order correction.

Therefore, by Eq.(A.98), we see locally it's not much different from the unconstrainted case Eq.(A.1) up to some negeligable correction. In the following analysis, we will always use formula Eq.(A.98) as the update equation for PSGD.

Since most of following proof bears a lot similarity as in unconstrained case, we only pointed out the essential steps in our following proof.

Theorem A.2.13 (Main Theorem for Equality-Constrained Case)**.** *Suppose a function $f(w) : \mathbb{R}^d \to \mathbb{R}$ with constraints $c_i(w) : \mathbb{R}^d \to \mathbb{R}$ is $(\lambda, \gamma, \varepsilon, \delta)$-strict saddle, and has a stochastic gradient oracle with radius at most Q, also satisfying $\mathbb{E}\xi = 0$ and $\mathbb{E}\xi\xi^T = \sigma^2 I$. Further, suppose the function function f is B-bounded, $\hat{L}$-Lipschitz, L-smooth, and has ρ-Lipschitz Hessian, and the constraints $\{c_i\}_{i=1}^m$ is $\hat{L}_i$-Lipschitz, L_i-smooth, and has ρ_i-Lipschitz Hessian. Then there exists a threshold $\eta_{\max} = \tilde{\Theta}(1)$, so that for any $\zeta > 0$, and for any $\eta \le \eta_{\max} / \max\{1, \log(1/\zeta)\}$, with probability at least $1 - \zeta$ in $t = \tilde{O}(\eta^{-2} \log(1/\zeta))$ iterations, PSGD outputs a point w_t that is $\tilde{O}(\sqrt{\eta \log(1/\eta\zeta)})$-close to some local minimum $w^\star$.*

First, we proof the assumptions in main theorem implies the smoothness conditions for $\mathfrak{M}(w)$, $\mathfrak{N}(w)$ and $\nabla^2_{ww}\mathcal{L}(w, \lambda^*(w'))$.

Lemma A.2.14. *Under the assumptions of Theorem A.2.13, there exists $L_M, L_N, \rho_M, \rho_N, \rho_L$ polynomial related to $B, \hat{L}, L, \rho, \frac{1}{\alpha_c}$ and $\{\hat{L}_i, L_i, \rho_i\}_{i=1}^m$ so that:*

1. *$\|\mathfrak{M}(w)\| \le L_M$ and $\|\mathfrak{N}(w)\| \le L_N$ for all $w \in \mathcal{W}$.*
2. *$\mathfrak{M}(w)$ is ρ_M-Lipschitz, and $\mathfrak{N}(w)$ is ρ_N-Lipschitz, and $\nabla^2_{ww}\mathcal{L}(w, \lambda^*(w'))$ is ρ_L-Lipschitz for all $w' \in \mathcal{W}$.*

Proof. By definition of $\mathfrak{M}(w)$, $\mathfrak{N}(w)$ and $\nabla^2_{ww}\mathcal{L}(w, \lambda^*(w'))$, the above conditions will holds if there exists $B_\lambda, \hat{L}_\lambda, L_\lambda$ bounded by $\tilde{O}(1)$, so that $\lambda^*(w)$ is B_λ-bounded, $\hat{L}_\lambda$-Lipschitz, and L_λ-smooth.

By definition Eq.(A.68), we have:

$$\lambda^*(w) = C(w)^\dagger \nabla f(w) = (C(w)^T C(w))^{-1} C(w)^T \nabla f(w) \tag{A.99}$$

Because f is B-bounded, $\hat{L}$-Lipschitz, L-smooth, and its Hessian is ρ-Lipschitz, thus, eventually, we only need to prove that there exists $B_c, \hat{L}_c, L_c$ bounded by $\tilde{O}(1)$, so that the pseudo-inverse $C(w)^\dagger$ is B_c-bounded, $\hat{L}_c$-Lipschitz, and L_c-smooth.

Since α_c-RLICQ holds for all feasible points, we immediately have: $\|C(w)^\dagger\| \leq \frac{1}{\alpha_c}$, thus bounded. For simplicity, in the following context we use $C^\dagger$ to represent $C^\dagger(w)$ without ambiguity. By some calculation of linear algebra, we have the derivative of pseudo-inverse:

$$\frac{\partial C(w)^\dagger}{\partial w_i} = -C^\dagger \frac{\partial C(w)}{\partial w_i} C^\dagger + C^\dagger [C^\dagger]^T \frac{\partial C(w)^T}{\partial w_i} (I - CC^\dagger) \tag{A.100}$$

Again, α_c-RLICQ holds implies that derivative of pseudo-inverse is well-defined for every feasible point. Let tensor $E(w), \tilde{E}(w)$ to be the derivative of $C(w), C^\dagger(w)$, which is defined as:

$$[E(w)]_{ijk} = \frac{\partial [C(w)]_{ik}}{\partial w_j} \qquad [\tilde{E}(w)]_{ijk} = \frac{\partial [C(w)^\dagger]_{ik}}{\partial w_j} \tag{A.101}$$

Define the transpose of a 3rd order tensor $E^T_{i,j,k} = E_{k,j,i}$, then we have

$$\tilde{E}(w) = -[E(w)](C^\dagger, I, C^\dagger) + [E(w)^T](C^\dagger [C^\dagger]^T, I, (I - CC^\dagger)) \tag{A.102}$$

where by calculation $[E(w)](I, I, e_i) = \nabla^2 c_i(w)$.

Finally, since $C(w)^\dagger$ and $\nabla^2 c_i(w)$ are bounded by $\tilde{O}(1)$, by Eq.(A.102), we know $\tilde{E}(w)$ is bounded, that is $C(w)^\dagger$ is Lipschitz. Again, since both $C(w)^\dagger$ and $\nabla^2 c_i(w)$ are bounded, Lipschitz, by Eq.(A.102), we know $\tilde{E}(w)$ is also $\tilde{O}(1)$-Lipschitz. This finishes the proof.

□

From now on, we can use the same proof strategy as unconstraint case. Below we list the corresponding lemmas and the essential steps that require modifications.

Lemma A.2.15. *Under the assumptions of Theorem A.2.13, with notations in Lemma A.2.14, for any point with* $\|\chi(w_0)\| \geq \sqrt{2\eta\sigma^2 L_M(d-m)}$ *where* $\sqrt{2\eta\sigma^2 L_M(d-m)} < \varepsilon$*, after one iteration we have:*

$$\mathbb{E}f(w_1) - f(w_0) \leq -\tilde{\Omega}(\eta^2) \tag{A.103}$$

Proof. Choose $\eta_{\max} < \frac{1}{L_M}$, and also small enough, then by update equation Eq.(A.98), we have:

$$\begin{aligned}
\mathbb{E}f(w_1) - f(w_0) &\leq \chi(w_0)^T \mathbb{E}(w_1 - w_0) + \frac{L_M}{2}\mathbb{E}\|w_1 - w_0\|^2 \\
&\leq -(\eta - \frac{L_M\eta^2}{2})\|\chi(w_0)\|^2 + \frac{\eta^2\sigma^2 L_M(d-m)}{2} + \tilde{O}(\eta^2)\|\chi(w_0)\| + \tilde{O}(\eta^3) \\
&\leq -(\eta - \tilde{O}(\eta^{1.5}) - \frac{L_M\eta^2}{2})\|\chi(w_0)\|^2 + \frac{\eta^2\sigma^2 L_M(d-m)}{2} + \tilde{O}(\eta^3) \\
&\leq -\frac{\eta^2\sigma^2 L_M d}{4}
\end{aligned} \tag{A.104}$$

Which finishes the proof. □

Theorem A.2.16. *Under the assumptions of Theorem A.2.13, with notations in Lemma A.2.14, for any initial point* w_0 *that is* $\tilde{O}(\sqrt{\eta}) < \delta$ *close to a local minimum* $w^\star$*, with probability at least* $1 - \zeta/2$*, we have following holds simultaneously:*

$$\forall t \leq \tilde{O}(\frac{1}{\eta^2}\log\frac{1}{\zeta}), \quad \|w_t - w^\star\| \leq \tilde{O}(\sqrt{\eta\log\frac{1}{\eta\zeta}}) < \delta \tag{A.105}$$

where $w^\star$ *is the locally optimal point.*

Proof. By calculus, we know

$$\chi(w_t) = \chi(w^\star) + \int_0^1 (\mathfrak{M} + \mathfrak{N})(w^\star + t(w_t - w^\star))\mathrm{d}t \cdot (w_t - w^\star) \tag{A.106}$$

Let filtration $\mathfrak{F}_t = \sigma\{\xi_0, \cdots \xi_{t-1}\}$, and note $\sigma\{\Delta_0, \cdots, \Delta_t\} \subset \mathfrak{F}_t$, where $\sigma\{\cdot\}$ denotes the sigma field. Let event $\mathfrak{E}_t = \{\forall\tau \leq t, \|w_\tau - w^\star\| \leq \mu\sqrt{\eta\log\frac{1}{\eta\zeta}} < \delta\}$, where μ is independent of (η, ζ), and will be specified later.

By Definition A.2.11 of $(\lambda, \gamma, \varepsilon, \delta)$-strict saddle, we know $\mathfrak{M}(w)$ is locally λ-strongly convex restricted to its tangent space $\mathcal{T}(w)$. in the 2δ-neighborhood of $w^\star$. If $\eta_{\max}$ is chosen small enough, by Remark 4 and Lemma A.2.7, we have in addition:

$$\begin{aligned}\chi(w_t)^T(w_t - w^\star)1_{\mathfrak{E}_t} &= (w_t - w^\star)^T \int_0^1 (\mathfrak{M} + \mathfrak{N})(w^\star + t(w_t - w^\star))\mathrm{d}t \cdot (w_t - w^\star)1_{\mathfrak{E}_t} \\ &\geq [\lambda\|w_t - w^\star\|^2 - \tilde{O}(\|w_t - w^\star\|^3)]1_{\mathfrak{E}_t} \geq 0.5\lambda\|w_t - w^\star\|^2 1_{\mathfrak{E}_t} \end{aligned} \tag{A.107}$$

Then, everything else follows almost the same as the proof of Lemma A.1.3. □

Lemma A.2.17. *Under the assumptions of Theorem A.2.13, with notations in Lemma A.2.14, for any initial point w_0 where $\|\chi(w_0)\| \leq \tilde{O}(\eta) < \varepsilon$, and $\hat{v}^T\mathfrak{M}(w_0)\hat{v} \leq -\gamma$ for some $\hat{v} \in \mathcal{T}(w)$, $\|\hat{v}\| = 1$, then there is a number of steps T that depends on w_0 such that:*

$$\mathbb{E}f(w_T) - f(w_0) \leq -\tilde{\Omega}(\eta) \tag{A.108}$$

The number of steps T has a fixed upper bound T_{max} that is independent of w_0 where $T \leq T_{max} = O((\log(d-m))/\gamma\eta)$.

Similar to the unconstrained case, we show this by a coupling sequence. Here the sequence we construct will only walk on the tangent space, by Lemmas in previous subsection, we know this is not very far from the actual sequence. We first define and characterize the coupled sequence in the following lemma:

Lemma A.2.18. *Under the assumptions of Theorem A.2.13, with notations in Lemma A.2.14. Let $\tilde{f}$ defined as local second-order approximation of $f(x)$ around w_0 in tangent space $\mathcal{T}_0 = \mathcal{T}(w_0)$:*

$$\tilde{f}(w) \doteq f(w_0) + \chi(w_0)^T(w - w_0) + \frac{1}{2}(w - w_0)^T[P_{\mathcal{T}_0}^T\mathfrak{M}(w_0)P_{\mathcal{T}_0}](w - w_0) \tag{A.109}$$

$\{\tilde{w}_t\}$ be the corresponding sequence generated by running SGD on function $\tilde{f}$, with $\tilde{w}_0 = w_0$, and noise projected to $\mathcal{T}_0$, (i.e. $\tilde{w}_t = \tilde{w}_{t-1} - \eta(\tilde{\chi}(\tilde{w}_{t-1}) + P_{\mathcal{T}_0}\xi_{t-1})$. For simplicity, denote $\tilde{\chi}(w) = \nabla\tilde{f}(w)$,

and $\widetilde{\mathfrak{M}} = P_{\mathcal{T}_0}^T \mathfrak{M}(w_0) P_{\mathcal{T}_0}$*, then we have analytically:*

$$\tilde{\chi}(\tilde{w}_t) = (1 - \eta\widetilde{\mathfrak{M}})^t \tilde{\chi}(\tilde{w}_0) - \eta\widetilde{\mathfrak{M}} \sum_{\tau=0}^{t-1} (1 - \eta\widetilde{\mathfrak{M}})^{t-\tau-1} P_{\mathcal{T}_0} \xi_\tau \tag{A.110}$$

$$\tilde{w}_t - w_0 = -\eta \sum_{\tau=0}^{t-1} (1 - \eta\widetilde{\mathfrak{M}})^\tau \tilde{\chi}(\tilde{w}_0) - \eta \sum_{\tau=0}^{t-1} (1 - \eta\widetilde{\mathfrak{M}})^{t-\tau-1} P_{\mathcal{T}_0} \xi_\tau \tag{A.111}$$

Furthermore, for any initial point w_0 *where* $\|\chi(w_0)\| \le \tilde{O}(\eta) < \varepsilon$*, and* $\min_{\hat{v}\in\mathcal{T}(w),\|\hat{v}\|=1} \hat{v}^T \mathfrak{M}(w_0)\hat{v} = -\gamma_0$*. There exist a* $T \in \mathbb{N}$ *satisfying:*

$$\frac{d-m}{\eta\gamma_0} \le \sum_{\tau=0}^{T-1} (1 + \eta\gamma_0)^{2\tau} < \frac{3(d-m)}{\eta\gamma_0} \tag{A.112}$$

with probability at least $1 - \tilde{O}(\eta^3)$*, we have following holds simultaneously for all* $t \le T$*:*

$$\|\tilde{w}_t - w_0\| \le \tilde{O}(\eta^{\frac{1}{2}} \log \frac{1}{\eta}); \qquad \|\tilde{\chi}(\tilde{w}_t)\| \le \tilde{O}(\eta^{\frac{1}{2}} \log \frac{1}{\eta}) \tag{A.113}$$

Proof. Clearly we have:

$$\tilde{\chi}(\tilde{w}_t) = \tilde{\chi}(\tilde{w}_{t-1}) + \widetilde{\mathfrak{M}}(\tilde{w}_t - \tilde{w}_{t-1}) \tag{A.114}$$

and

$$\tilde{w}_t = \tilde{w}_{t-1} - \eta(\tilde{\chi}(\tilde{w}_{t-1}) + P_{\mathcal{T}_0}\xi_{t-1}) \tag{A.115}$$

This lemma is then proved by a direct application of Lemma A.1.5. □

Then we show the sequence constructed is very close to the actual sequence.

Lemma A.2.19. *Under the assumptions of Theorem A.2.13, with notations in Lemma A.2.14. Let* $\{w_t\}$ *be the corresponding sequence generated by running PSGD on function* f*. Also let* $\tilde{f}$ *and* $\{\tilde{w}_t\}$ *be defined as in Lemma A.2.18. Then, for any initial point* w_0 *where* $\|\chi(w_0)\|^2 \le \tilde{O}(\eta) < \varepsilon$*, and* $\min_{\hat{v}\in\mathcal{T}(w),\|\hat{v}\|=1} \hat{v}^T \mathfrak{M}(w_0)\hat{v} = -\gamma_0$*. Given the choice of* T *as in Eq.(A.112), with probability at least* $1 - \tilde{O}(\eta^2)$*, we have following holds simultaneously for all* $t \le T$*:*

$$\|w_t - \tilde{w}_t\| \le \tilde{O}(\eta \log^2 \frac{1}{\eta}); \tag{A.116}$$

Proof. First, we have update function of tangent gradient by:

$$\begin{aligned}\chi(w_t) =&\chi(w_{t-1}) + \int_0^1 \nabla\chi(w_{t-1} + t(w_t - w_{t-1}))\mathrm{d}t \cdot (w_t - w_{t-1}) \\ =&\chi(w_{t-1}) + \mathfrak{M}(w_{t-1})(w_t - w_{t-1}) + \mathfrak{N}(w_{t-1})(w_t - w_{t-1}) + \theta_{t-1}\end{aligned} \tag{A.117}$$

where the remainder:

$$\theta_{t-1} \equiv \int_0^1 \left[\nabla\chi(w_{t-1} + t(w_t - w_{t-1})) - \nabla\chi(w_{t-1})\right] \mathrm{d}t \cdot (w_t - w_{t-1}) \tag{A.118}$$

Project it to tangent space $\mathcal{T}_0 = \mathcal{T}(w_0)$. Denote $\widetilde{\mathfrak{M}} = P_{\mathcal{T}_0}^T \mathfrak{M}(w_0) P_{\mathcal{T}_0}$, and $\widetilde{\mathfrak{M}}'_{t-1} = P_{\mathcal{T}_0}^T [\, \mathfrak{M}(w_{t_1}) - \mathfrak{M}(w_0)\,] P_{\mathcal{T}_0}$. Then, we have:

$$\begin{aligned}P_{\mathcal{T}_0} \cdot \chi(w_t) =&P_{\mathcal{T}_0} \cdot \chi(w_{t-1}) + P_{\mathcal{T}_0}(\mathfrak{M}(w_{t-1}) + \mathfrak{N}(w_{t-1}))(w_t - w_{t-1}) + P_{\mathcal{T}_0}\theta_{t-1} \\ =&P_{\mathcal{T}_0} \cdot \chi(w_{t-1}) + P_{\mathcal{T}_0}\mathfrak{M}(w_{t-1})P_{\mathcal{T}_0}(w_t - w_{t-1}) \\ &+ P_{\mathcal{T}_0}\mathfrak{M}(w_{t-1})P_{\mathcal{T}_0^c}(w_t - w_{t-1}) + P_{\mathcal{T}_0}\mathfrak{N}(w_{t-1})(w_t - w_{t-1}) + P_{\mathcal{T}_0}\theta_{t-1} \\ =&P_{\mathcal{T}_0} \cdot \chi(w_{t-1}) + \widetilde{\mathfrak{M}}(w_t - w_{t-1}) + \phi_{t-1}\end{aligned} \tag{A.119}$$

Where

$$\phi_{t-1} = [\, \widetilde{\mathfrak{M}}'_{t-1} + P_{\mathcal{T}_0}\mathfrak{M}(w_{t-1})P_{\mathcal{T}_0^c} + P_{\mathcal{T}_0}\mathfrak{N}(w_{t-1})\,](w_t - w_{t-1}) + P_{\mathcal{T}_0}\theta_{t-1} \tag{A.120}$$

By Hessian smoothness, we immediately have:

$$\|\widetilde{\mathfrak{M}}'_{t-1}\| = \|\mathfrak{M}(w_{t_1}) - \mathfrak{M}(w_0)\| \le \rho_M \|w_{t-1} - w_0\| \le \rho_M(\|w_t - \tilde{w}_t\| + \|\tilde{w}_t - w_0\|) \tag{A.121}$$

$$\|\theta_{t-1}\| \le \frac{\rho_M + \rho_N}{2}\|w_t - w_{t-1}\|^2 \tag{A.122}$$

Substitute the update equation of PSGD (Eq.(A.98)) into Eq.(A.119), we have:

$$\begin{aligned}P_{\mathcal{T}_0} \cdot \chi(w_t) &= P_{\mathcal{T}_0} \cdot \chi(w_{t-1}) - \eta\widetilde{\mathfrak{M}}(P_{\mathcal{T}_0} \cdot \chi(w_{t-1}) + P_{\mathcal{T}_0} \cdot P_{\mathcal{T}(w_{t-1})}\xi_{t-1}) + \widetilde{\mathfrak{M}} \cdot \iota_{t-1} + \phi_{t-1} \\ &= (1 - \eta\widetilde{\mathfrak{M}})P_{\mathcal{T}_0} \cdot \chi(w_{t-1}) - \eta\widetilde{\mathfrak{M}}P_{\mathcal{T}_0}\xi_{t-1} + \eta\widetilde{\mathfrak{M}}P_{\mathcal{T}_0} \cdot P_{\mathcal{T}^c(w_{t-1})}\xi_{t-1} + \widetilde{\mathfrak{M}} \cdot \iota_{t-1} + \phi_{t-1}\end{aligned} \tag{A.123}$$

Let $\Delta_t = P_{\mathcal{T}_0} \cdot \chi(w_t) - \tilde{\chi}(\tilde{w}_t)$ denote the difference of tangent gradient in $\mathcal{T}(w_0)$, then from Eq.(A.114), Eq.(A.115), and Eq.(A.123) we have:

$$\Delta_t = (1 - \eta H)\Delta_{t-1} + \eta \widetilde{\mathfrak{M}} P_{\mathcal{T}_0} \cdot P_{\mathcal{T}^c(w_{t-1})}\xi_{t-1} + \widetilde{\mathfrak{M}} \cdot \iota_{t-1} + \phi_{t-1} \tag{A.124}$$

$$P_{\mathcal{T}_0} \cdot (w_t - w_0) - (\tilde{w}_t - w_0) = -\eta \sum_{\tau=0}^{t-1} \Delta_\tau + \eta \sum_{\tau=0}^{t-1} P_{\mathcal{T}_0} \cdot P_{\mathcal{T}^c(w_\tau)}\xi_\tau + \sum_{\tau=0}^{t-1} \iota_\tau \tag{A.125}$$

By Lemma A.2.7, we know if $\sum_{i=1}^m \frac{L_i^2}{a_c^2} = \frac{1}{R^2}$, then we have:

$$\|P_{\mathcal{T}_0^c}(w_t - w_0)\| \leq \frac{\|w_t - w_0\|^2}{2R} \tag{A.126}$$

Let filtration $\mathfrak{F}_t = \sigma\{\xi_0, \cdots \xi_{t-1}\}$, and note $\sigma\{\Delta_0, \cdots, \Delta_t\} \subset \mathfrak{F}_t$, where $\sigma\{\cdot\}$ denotes the sigma field. Also, let event $\mathfrak{K}_t = \{\forall \tau \leq t,\ \|\tilde{\chi}(\tilde{w}_\tau)\| \leq \tilde{O}(\eta^{\frac{1}{2}} \log \frac{1}{\eta}),\ \|\tilde{w}_\tau - w_0\| \leq \tilde{O}(\eta^{\frac{1}{2}} \log \frac{1}{\eta})\}$, and denote $\Gamma_t = \eta \sum_{\tau=0}^{t-1} P_{\mathcal{T}_0} \cdot P_{\mathcal{T}^c(w_\tau)}\xi_\tau$, let $\mathfrak{E}_t = \{\forall \tau \leq t,\ \|\Delta_\tau\| \leq \mu_1 \eta \log^2 \frac{1}{\eta}, \|\Gamma_\tau\| \leq \mu_2 \eta \log^2 \frac{1}{\eta}, \|w_\tau - \tilde{w}_\tau\| \leq \mu_3 \eta \log^2 \frac{1}{\eta}\}$ where (μ_1, μ_2, μ_3) are is independent of (η, ζ), and will be determined later. To prevent ambiguity in the proof, $\tilde{O}$ notation will not hide any dependence on μ. Clearly event $\mathfrak{K}_{t-1} \subset \mathfrak{F}_{t-1}, \mathfrak{E}_{t-1} \subset \mathfrak{F}_{t-1}$ thus independent of ξ_{t-1}.

Then, conditioned on event $\mathfrak{K}_{t-1} \cap \mathfrak{E}_{t-1}$, by triangle inequality, we have $\|w_\tau - w_0\| \leq \tilde{O}(\eta^{\frac{1}{2}} \log \frac{1}{\eta})$, for all $\tau \leq t - 1 \leq T - 1$. We then need to carefully bound the following bound each term in Eq.(A.124). We know $w_t - w_{t-1} = -\eta \cdot (\chi(w_{t-1}) + P_{\mathcal{T}(w_{t-1})}\xi_{t-1}) + \iota_{t-1}$, and then by Lemma A.2.9 and Lemma A.2.8, we have:

$$\|\eta \widetilde{\mathfrak{M}} P_{\mathcal{T}_0} \cdot P_{\mathcal{T}^c(w_{t-1})}\xi_{t-1}\| \leq \tilde{O}(\eta^{1.5} \log \frac{1}{\eta})$$

$$\|\widetilde{\mathfrak{M}} \cdot \iota_{t-1}\| \leq \tilde{O}(\eta^2)$$

$$\|[\ \widetilde{\mathfrak{M}}'_{t-1} + P_{\mathcal{T}_0}\mathfrak{M}(w_{t-1})P_{\mathcal{T}_0^c} + P_{\mathcal{T}_0}\mathfrak{N}(w_{t-1})\ ](-\eta \cdot \chi(w_{t-1}))\| \leq \tilde{O}(\eta^2 \log^2 \frac{1}{\eta})$$

$$\|[\ \widetilde{\mathfrak{M}}'_{t-1} + P_{\mathcal{T}_0}\mathfrak{M}(w_{t-1})P_{\mathcal{T}_0^c} + P_{\mathcal{T}_0}\mathfrak{N}(w_{t-1})\ ](-\eta P_{\mathcal{T}(w_{t-1})}\xi_{t-1})\| \leq \tilde{O}(\eta^{1.5} \log \frac{1}{\eta})$$

$$\|[\ \widetilde{\mathfrak{M}}'_{t-1} + P_{\mathcal{T}_0}\mathfrak{M}(w_{t-1})P_{\mathcal{T}_0^c} + P_{\mathcal{T}_0}\mathfrak{N}(w_{t-1})\ ]\iota_{t-1}\| \leq \tilde{O}(\eta^2)$$

$$\|P_{\mathcal{T}_0}\theta_{t-1}\| \le \tilde{O}(\eta^2) \tag{A.127}$$

Therefore, abstractly, conditioned on event $\mathfrak{K}_{t-1} \cap \mathfrak{E}_{t-1}$, we could write down the recursive equation as:

$$\Delta_t = (1 - \eta H)\Delta_{t-1} + A + B \tag{A.128}$$

where $\|A\| \le \tilde{O}(\eta^{1.5} \log \frac{1}{\eta})$ and $\|B\| \le \tilde{O}(\eta^2 \log^2 \frac{1}{\eta})$, and in addition, by independence, easy to check we also have $\mathbb{E}[(1 - \eta H)\Delta_{t-1}A|\mathfrak{F}_{t-1}] = 0$. This is exactly the same case as in the proof of Lemma A.1.6. By the same argument of martingale and Azuma-Hoeffding, and by choosing μ_1 large enough, we can prove

$$P\left(\mathfrak{E}_{t-1} \cap \left\{\|\Delta_t\| \ge \mu_1 \eta \log^2 \frac{1}{\eta}\right\}\right) \le \tilde{O}(\eta^3) \tag{A.129}$$

On the other hand, for $\Gamma_t = \eta \sum_{\tau=0}^{t-1} P_{\mathcal{T}_0} \cdot P_{\mathcal{T}^c(w_\tau)}\xi_\tau$, we have:

$$\begin{aligned}\mathbb{E}[\Gamma_t 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}}|\mathfrak{F}_{t-1}] &= \left[\Gamma_{t-1} + \eta\mathbb{E}[P_{\mathcal{T}_0} \cdot P_{\mathcal{T}^c(w_{t-1})}\xi_{t-1}|\mathfrak{F}_{t-1}]\right] 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} \\ &= \Gamma_{t-1}1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} \le \Gamma_{t-1}1_{\mathfrak{K}_{t-2}\cap\mathfrak{E}_{t-2}}\end{aligned} \tag{A.130}$$

Therefore, we have $\mathbb{E}[\Gamma_t 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} \mid \mathfrak{F}_{t-1}] \le \Gamma_{t-1}1_{\mathfrak{K}_{t-2}\cap\mathfrak{E}_{t-2}}$ which means $\Gamma_t 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}}$ is a supermartingale.

We also know by Lemma A.2.9, with probability 1:

$$\begin{aligned}&|\Gamma_t 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} - \mathbb{E}[\Gamma_t 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} \mid \mathfrak{F}_{t-1}]| = |\eta P_{\mathcal{T}_0} \cdot P_{\mathcal{T}^c(w_{t-1})}\xi_{t-1}| \cdot 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} \\ \le&\tilde{O}(\eta)\|w_{t-1} - w_0\|1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} \le \tilde{O}(\eta^{1.5}\log\frac{1}{\eta}) = c_{t-1}\end{aligned} \tag{A.131}$$

By Azuma-Hoeffding inequality, with probability less than $\tilde{O}(\eta^3)$, for $t \le T \le O(\log(d-m)/\gamma_0\eta)$:

$$\Gamma_t 1_{\mathfrak{K}_{t-1}\cap\mathfrak{E}_{t-1}} - \Gamma_0 \cdot 1 > \tilde{O}(1)\sqrt{\sum_{\tau=0}^{t-1} c_\tau^2 \log(\frac{1}{\eta})} = \tilde{O}(\eta \log^2 \frac{1}{\eta}) \tag{A.132}$$

This means there exists some $\tilde{C}_2 = \tilde{O}(1)$ so that:

$$P\left(\Re_{t-1} \cap \mathfrak{E}_{t-1} \cap \left\{\|\Gamma_t\| \geq \tilde{C}_2 \eta \log^2 \frac{1}{\eta}\right\}\right) \leq \tilde{O}(\eta^3) \tag{A.133}$$

by choosing $\mu_2 > \tilde{C}_2$, we have:

$$P\left(\Re_{t-1} \cap \mathfrak{E}_{t-1} \cap \left\{\|\Gamma_t\| \geq \mu_2 \eta \log^2 \frac{1}{\eta}\right\}\right) \leq \tilde{O}(\eta^3) \tag{A.134}$$

Therefore, combined with Lemma A.2.18, we have:

$$P\left(\mathfrak{E}_{t-1} \cap \left\{\|\Gamma_t\| \geq \mu_2 \eta \log^2 \frac{1}{\eta}\right\}\right) \leq \tilde{O}(\eta^3) + P(\overline{\Re}_{t-1}) \leq \tilde{O}(\eta^3) \tag{A.135}$$

Finally, conditioned on event $\Re_{t-1} \cap \mathfrak{E}_{t-1}$, if we have $\|\Gamma_t\| \leq \mu_2 \eta \log^2 \frac{1}{\eta}$, then by Eq.(A.125):

$$\|P_{\mathcal{T}_0} \cdot (w_t - w_0) - (\tilde{w}_t - w_0)\| \leq \tilde{O}\left((\mu_1 + \mu_2)\eta \log^2 \frac{1}{\eta}\right) \tag{A.136}$$

Since $\|w_{t-1} - w_0\| \leq \tilde{O}(\eta^{\frac{1}{2}} \log \frac{1}{\eta})$, and $\|w_t - w_{t-1}\| \leq \tilde{O}(\eta)$, by Eq.(A.126):

$$\|P_{\mathcal{T}_0^c}(w_t - w_0)\| \leq \frac{\|w_t - w_0\|^2}{2R} \leq \tilde{O}(\eta \log^2 \frac{1}{\eta}) \tag{A.137}$$

Thus:

$$\begin{aligned} \|w_t - \tilde{w}_t\|^2 =& \|P_{\mathcal{T}_0} \cdot (w_t - \tilde{w}_t)\|^2 + \|P_{\mathcal{T}_0^c} \cdot (w_t - \tilde{w}_t)\|^2 \\ =& \|P_{\mathcal{T}_0} \cdot (w_t - w_0) - (\tilde{w}_t - w_0)\|^2 + \|P_{\mathcal{T}_0^c}(w_t - w_0)\|^2 \leq \tilde{O}((\mu_1 + \mu_2)^2 \eta^2 \log^4 \frac{1}{\eta}) \end{aligned} \tag{A.138}$$

That is there exist some $\tilde{C}_3 = \tilde{O}(1)$ so that $\|w_t - \tilde{w}_t\| \leq \tilde{C}_3(\mu_1 + \mu_2)\eta \log^2 \frac{1}{\eta}$ Therefore, conditioned on event $\Re_{t-1} \cap \mathfrak{E}_{t-1}$, we have proved that if choose $\mu_3 > \tilde{C}_3(\mu_1 + \mu_2)$, then event $\{\|w_t - \tilde{w}_t\| \geq \mu_3 \eta \log^2 \frac{1}{\eta}\} \subset \{\|\Gamma_t\| \geq \mu_2 \eta \log^2 \frac{1}{\eta}\}$. Then, combined this fact with Eq.(A.129), Eq.(A.135), we have proved:

$$P\left(\mathfrak{E}_{t-1} \cap \overline{\mathfrak{E}}_t\right) \leq \tilde{O}(\eta^3) \tag{A.139}$$

Because $P(\overline{\mathfrak{E}}_0) = 0$, and $T \leq \tilde{O}(\frac{1}{\eta})$, we have $P(\overline{\mathfrak{E}}_T) \leq \tilde{O}(\eta^2)$, which concludes the proof.

□

These two lemmas allow us to prove the result when the initial point is very close to a saddle point.

Proof of Lemma A.2.17. Combine Talyor expansion Eq.A.73 with Lemma A.2.18, Lemma A.2.19, we prove this Lemma by the same argument as in the proof of Lemma A.1.4. □

Finally the main theorem follows.

Proof of Theorem A.2.13. By Lemma A.2.15, Lemma A.2.17, and Lemma A.2.16, with the same argument as in the proof Theorem A.1.1, we easily concludes this proof. □

A.3 Detailed proofs for Section 3.3

In this section we show two optimization problems (3.7) and (3.9) satisfy the $(\lambda, \gamma, \varepsilon, \delta)$-strict saddle propery.

A.3.1 Warm up: maximum eigenvalue formulation

Recall that we are trying to solve the optimization (3.7), which we restate here.

$$\begin{aligned} \max \quad & T(u,u,u,u), \\ & \|u\|^2 = 1. \end{aligned} \tag{A.140}$$

Here the tensor T has orthogonal decomposition $T = \sum_{i=1}^{d} a_i^{\otimes 4}$. We first do a change of coordinates to work in the coordinate system specified by (a_i)'s (this does not change the dynamics of the algorithm). In particular, let $u = \sum_{i=1}^{d} x_i a_i$ (where $x \in \mathbb{R}^d$), then we can see $T(u,u,u,u) = \sum_{i=1}^{d} x_i^4$. Therefore let $f(x) = -\|x\|_4^4$, the optimization problem is equivalent to

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & \|x\|_2^2 = 1 \end{aligned} \tag{A.141}$$

This is a constrained optimization, so we apply the framework developed in Section 3.2.3.

Let $c(x) = \|x\|_2^2 - 1$. We first compute the Lagrangian

$$\mathcal{L}(x, \lambda) = f(x) - \lambda c(x) = -\|x\|_4^4 - \lambda(\|x\|_2^2 - 1). \tag{A.142}$$

Since there is only one constraint, and the gradient when $\|x\| = 1$ always have norm 2, we know the set of constraints satisfy 2-RLICQ. In particular, we can compute the correct value of Lagrangian multiplier λ,

$$\lambda^*(x) = \arg\min_{\lambda} \|\nabla_x \mathcal{L}(x,\lambda)\| = \arg\min_{\lambda} \sum_{i=1}^{d} (2x_i^3 + \lambda x_i)^2 = -2\|x\|_4^4 \tag{A.143}$$

Therefore, the gradient in the tangent space is equal to

$$\begin{aligned}\chi(x) &= \nabla_x \mathcal{L}(x,\lambda)|_{(x,\lambda^*(x))} = \nabla f(x) - \lambda^*(x)\nabla c(x) \\ &= -4(x_1^3, \cdots, x_d^3)^T - 2\lambda^*(x)(x_1, \cdots, x_d)^T \\ &= 4\left((x_1^2 - \|x\|_4^4)x_1, \cdots, (x_d^2 - \|x\|_4^4)x_d\right)\end{aligned} \tag{A.144}$$

The second-order partial derivative of Lagrangian is equal to

$$\begin{aligned}\mathfrak{M}(x) &= \nabla_{xx}^2 \mathcal{L}(x,\lambda)|_{(x,\lambda^*(x))} = \nabla^2 f(x) - \lambda^*(x)\nabla^2 c(x) \\ &= -12\mathrm{diag}(x_1^2, \cdots, x_d^2) - 2\lambda^*(x) I_d \\ &= -12\mathrm{diag}(x_1^2, \cdots, x_d^2) + 4\|x\|_4^4 I_d\end{aligned} \tag{A.145}$$

Since the variable x has bounded norm, and the function is a polynomial, it's clear that the function itself is bounded and all its derivatives are bounded. Moreover, all the derivatives of the constraint are bounded. We summarize this in the following lemma.

Lemma A.3.1. *The objective function (3.7) is bounded by* 1*, its p-th order derivative is bounded by* $O(\sqrt{d})$ *for* $p = 1, 2, 3$*. The constraint's p-th order derivative is bounded by* 2*, for* $p = 1, 2, 3$*.*

Therefore the function satisfy all the smoothness condition we need. Finally we show the gradient and Hessian of Lagrangian satisfy the $(\lambda, \gamma, \varepsilon, \delta)$-strict saddle property. Note that we did not try to optimize the dependency with respect to d.

Theorem A.3.2. *The only local minima of optimization problem (3.7) are* $\pm a_i$ $(i \in [d])$*. Further it satisfy* $(\lambda, \gamma, \varepsilon, \delta)$*-strict saddle for* $\gamma = 7/d$*,* $\lambda = 3$ *and* $\varepsilon, \delta = 1/poly(d)$*.*

In order to prove this theorem, we consider the transformed version Eq.A.141. We first need following two lemma for points around saddle point and local minimum respectively. We choose

$$\varepsilon_0 = (10d)^{-4}, \;\; \varepsilon = 4\varepsilon_0^2, \;\; \delta = 2d\varepsilon_0, \;\; \mathfrak{S}(x) = \{i \mid |x_i| > \varepsilon_0\} \tag{A.146}$$

Where by intuition, $\mathfrak{S}(x)$ is the set of coordinates whose value is relative large.

Lemma A.3.3. *Under the choice of parameters in Eq.(A.146), suppose $\|\chi(x)\| \le \varepsilon$, and $|\mathfrak{S}(x)| \ge 2$. Then, there exists $\hat{v} \in \mathcal{T}(x)$ and $\|\hat{v}\| = 1$, so that $\hat{v}^T \mathfrak{M}(x)\hat{v} \le -7/d$.*

Proof. Suppose $|\mathfrak{S}(x)| = p$, and $2 \le p \le d$. Since $\|\chi(x)\| \le \varepsilon = 4\varepsilon_0^2$, by Eq.(A.144), we have for each $i \in [d]$, $|[\chi(x)]_i| = 4|(x_i^2 - \|x\|_4^4)x_i| \le 4\varepsilon_0^2$. Therefore, we have:

$$\forall i \in \mathfrak{S}(x), \qquad |x_i^2 - \|x\|_4^4| \le \varepsilon_0 \tag{A.147}$$

and thus:

$$\begin{aligned} &|\|x\|_4^4 - \frac{1}{p}| = |\|x\|_4^4 - \frac{1}{p}\sum_i x_i^2| \\ \le &|\|x\|_4^4 - \frac{1}{p}\sum_{i\in\mathfrak{S}(x)} x_i^2| + |\frac{1}{p}\sum_{i\in[d]-\mathfrak{S}(x)} x_i^2| \le \varepsilon_0 + \frac{d-p}{p}\varepsilon_0^2 \le 2\varepsilon_0 \end{aligned} \tag{A.148}$$

Combined with Eq.A.147, this means:

$$\forall i \in \mathfrak{S}(x), \qquad |x_i^2 - \frac{1}{p}| \le 3\varepsilon_0 \tag{A.149}$$

Because of symmetry, WLOG we assume $\mathfrak{S}(x) = \{1, \cdots, p\}$. Since $|\mathfrak{S}(x)| \ge 2$, we can pick $\hat{v} = (a, b, 0, \cdots, 0)$. Here $a > 0, b < 0$, and $a^2 + b^2 = 1$. We pick a such that $ax_1 + bx_2 = 0$. The solution is the intersection of a radius 1 circle and a line which passes $(0, 0)$, which always exists. For this $\hat{v}$, we know $\|\hat{v}\| = 1$, and $\hat{v}^T x = 0$ thus $\hat{v} \in \mathcal{T}(x)$. We have:

$$\hat{v}^T \mathfrak{M}(x)\hat{v} = -(12x_1^2 + 4\|x\|_4^4)a^2 - (12x_2^2 + 4\|x\|_4^4)b^2$$

$$
\begin{aligned}
&= -8x_1^2a^2 - 8x_2^2b^2 - 4(x_1^2 - \|x\|_4^4))a^2 - 4(x_2^2 - \|x\|_4^4))b^2 \\
&\leq -\frac{8}{p} + 24\varepsilon_0 + 4\varepsilon_0 \leq -7/d
\end{aligned} \tag{A.150}
$$

Which finishes the proof. □

Lemma A.3.4. *Under the choice of parameters in Eq.(A.146), suppose* $\|\chi(x)\| \leq \varepsilon$, *and* $|\mathfrak{S}(x)| = 1$. *Then, there is a local minimum* $x^\star$ *such that* $\|x - x^\star\| \leq \delta$, *and for all* x' *in the* 2δ *neighborhood of* $x^\star$, *we have* $\hat{v}^T\mathfrak{M}(x')\hat{v} \geq 3$ *for all* $\hat{v} \in \mathcal{T}(x')$, $\|\hat{v}\| = 1$

Proof. WLOG, we assume $\mathfrak{S}(x) = \{1\}$. Then, we immediately have for all $i > 1$, $|x_i| \leq \varepsilon_0$, and thus:

$$
1 \geq x_1^2 = 1 - \sum_{i>1} x_i^2 \geq 1 - d\varepsilon_0^2 \tag{A.151}
$$

Therefore $x_1 \geq \sqrt{1 - d\varepsilon_0^2}$ or $x_1 \leq -\sqrt{1 - d\varepsilon_0^2}$. Which means x_1 is either close to 1 or close to -1. By symmetry, we know WLOG, we can assume the case $x_1 \geq \sqrt{1 - d\varepsilon_0^2}$. Let $e_1 = (1, 0, \cdots, 0)$, then we know:

$$
\|x - e_1\|^2 \leq (x_1 - 1)^2 + \sum_{i>1} x_i^2 \leq 2d\varepsilon_0^2 \leq \delta^2 \tag{A.152}
$$

Next, we show e_1 is a local minimum. According to Eq.A.145, we know $\mathfrak{M}(e_1)$ is a diagonal matrix with 4 on the diagonals except for the first diagonal entry (which is equal to -8), since $\mathcal{T}(e_1) = \text{span}\{e_2, \cdots, e_d\}$, we have:

$$
v^T\mathfrak{M}(e_1)v \geq 4\|v\|^2 > 0 \qquad \text{for all } v \in \mathcal{T}(e_1), v \neq 0 \tag{A.153}
$$

Which by Theorem A.2.6 means e_1 is a local minimum.

Finally, denote $\mathcal{T}_1 = \mathcal{T}(e_1)$ be the tangent space of constraint manifold at e_1. We know for all x' in the 2δ neighborhood of e_1, and for all $\hat{v} \in \mathcal{T}(x')$, $\|\hat{v}\| = 1$:

$$
\hat{v}^T\mathfrak{M}(x')\hat{v} \geq \hat{v}^T\mathfrak{M}(e_1)\hat{v} - |\hat{v}^T\mathfrak{M}(e_1)\hat{v} - \hat{v}^T\mathfrak{M}(x')\hat{v}|
$$

$$
\begin{aligned}
&= 4\|P_{\mathcal{T}_1}\hat{v}\|^2 - 8\|P_{\mathcal{T}_1^c}\hat{v}\|^2 - \|\mathfrak{M}(e_1) - \mathfrak{M}(x')\|\|\hat{v}\|^2 \\
&= 4 - 12\|P_{\mathcal{T}_1^c}\hat{v}\|^2 - \|\mathfrak{M}(e_1) - \mathfrak{M}(x')\| \qquad \text{(A.154)}
\end{aligned}
$$

By lemma A.2.8, we know $\|P_{\mathcal{T}_1^c}\hat{v}\|^2 \le \|x' - e_1\|^2 \le 4\delta^2$. By Eq.(A.145), we have:

$$
\begin{aligned}
&\|\mathfrak{M}(e_1) - \mathfrak{M}(x')\| \le \|\mathfrak{M}(e_1) - \mathfrak{M}(x')\| \le \sum_{(i,j)} |[\mathfrak{M}(e_1)]_{ij} - [\mathfrak{M}(x')]_{ij}| \\
&\le \sum_i \left| -12[e_1]_i^2 + 4\|e_1\|_4^4 - 12x_i^2 + 4\|x\|_4^4 \right| \le 64d\delta \qquad \text{(A.155)}
\end{aligned}
$$

In conclusion, we have $\hat{v}^T\mathfrak{M}(x')\hat{v} \ge 4 - 48\delta^2 - 64d\delta \ge 3$ which finishs the proof. □

Finally, we are ready to prove Theorem A.3.2.

Proof of Theorem A.3.2. According to Lemma A.3.3 and Lemma A.3.4, we immediately know the optimization problem satisfies $(\lambda, \gamma, \varepsilon, \delta)$-strict saddle.

The only thing remains to show is that the only local minima of optimization problem (3.7) are $\pm a_i$ ($i \in [d]$). Which is equivalent to show that the only local minima of the transformed problem is $\pm e_i$ ($i \in [d]$), where $e_i = (0, \cdots, 0, 1, 0, \cdots, 0)$, where 1 is on i-th coordinate.

By investigating the proof of Lemma A.3.3 and Lemma A.3.4, we know these two lemmas actually hold for any small enough choice of ε_0 satisfying $\varepsilon_0 \le (10d)^{-4}$, by pushing $\varepsilon_0 \to 0$, we know for any point satisfying $|\chi(x)| \le \varepsilon \to 0$, if it is close to some local minimum, it must satisfy $1 = |\mathfrak{S}(x)| \to \mathrm{supp}(x)$. Therefore, we know the only possible local minima are $\pm e_i$ ($i \in [d]$). In Lemma A.3.4, we proved e_1 is local minimum, by symmetry, we finishes the proof. □

A.3.2 New formulation

In this section we consider our new formulation (3.9). We first restate the optimization problem here:

$$\begin{aligned} \min \quad & \sum_{i \neq j} T(u^{(i)}, u^{(i)}, u^{(j)}, u^{(j)}), \\ \forall i \quad & \|u^{(i)}\|^2 = 1. \end{aligned} \tag{A.156}$$

Note that we changed the notation for the variables from u_i to $u^{(i)}$, because in later proofs we will often refer to the particular coordinates of these vectors.

Similar to the previous section, we perform a change of basis. The effect is equivalent to making a_i's equal to basis vectors e_i (and hence the tensor is equal to $T = \sum_{i=1}^{d} e_i^{\otimes 4}$. After the transformation the equations become

$$\begin{aligned} \min \quad & \sum_{(i,j):i \neq j} h(u^{(i)}, u^{(j)}) \\ \text{s.t.} \quad & \|u^{(i)}\|^2 = 1 \qquad \forall i \in [d] \end{aligned} \tag{A.157}$$

Here $h(u^{(i)}, u^{(j)}) = \sum_{k=1}^{d} (u_k^{(i)} u_k^{(j)})^2$, $(i, j) \in [d]^2$. We divided the objective function by 2 to simplify the calculation.

Let $U \in \mathbb{R}^{d^2}$ be the concatenation of $\{u^{(i)}\}$ such that $U_{ij} = u_j^{(i)}$. Let $c_i(U) = \|u^{(i)}\|^2 - 1$ and $f(U) = \frac{1}{2} \sum_{(i,j):i \neq j} h(u^{(i)}, u^{(j)})$. We can then compute the Lagrangian

$$\mathcal{L}(U, \lambda) = f(U) - \sum_{i=1}^{d} \lambda_i c_i(U) = \frac{1}{2} \sum_{(i,j):i \neq j} h(u^{(i)}, u^{(j)}) - \sum_{i=1}^{d} \lambda_i (\|u^{(i)}\|^2 - 1) \tag{A.158}$$

The gradients of $c_i(U)$'s are equal to $(0, \cdots, 0, 2u^{(i)}, 0, \cdots, 0)^T$, all of these vectors are orthogonal

to each other (because they have disjoint supports) and have norm 2. Therefore the set of constraints satisfy 2-RLICQ. We can then compute the Lagrangian multipiers λ^* as follows

$$\lambda^*(U) = \arg\min_{\lambda} \|\nabla_U \mathcal{L}(U, \lambda)\| = \arg\min_{\lambda} 4 \sum_i \sum_k (\sum_{j:j\neq i} U_{jk}^2 U_{ik} - \lambda_i U_{ik})^2 \tag{A.159}$$

which gives:

$$\lambda_i^*(U) = \arg\min_{\lambda} \sum_k (\sum_{j:j\neq i} U_{jk}^2 U_{ik} - \lambda_i U_{ik})^2 = \sum_{j:j\neq i} h(u^{(j)}, u^{(i)}) \tag{A.160}$$

Therefore, gradient in the tangent space is equal to

$$\chi(U) = \nabla_U \mathcal{L}(U, \lambda)|_{(U,\lambda^*(U))} = \nabla f(U) - \sum_{i=1}^{n} \lambda_i^*(U) \nabla c_i(U). \tag{A.161}$$

The gradient is a d^2 dimensional vector (which can be viewed as a $d \times d$ matrix corresponding to entries of U), and we express this in a coordinate-by-coordinate way. For simplicity of later proof, denote:

$$\psi_{ik}(U) = \sum_{j:j\neq i} [U_{jk}^2 - h(u^{(j)}, u^{(i)})] = \sum_{j:j\neq i} [U_{jk}^2 - \sum_{l=1}^{d} U_{il}^2 U_{jl}^2] \tag{A.162}$$

Then we have:

$$\begin{aligned} [\chi(U)]_{ik} &= 2(\sum_{j:j\neq i} U_{jk}^2 - \lambda_i^*(U)) U_{ik} \\ &= 2U_{ik} \sum_{j:j\neq i} (U_{jk}^2 - h(u^{(j)}, u^{(i)})) \\ &= 2U_{ik} \psi_{ik}(U) \end{aligned} \tag{A.163}$$

Similarly we can compute the second-order partial derivative of Lagrangian as

$$\mathfrak{M}(U) = \nabla^2 f(U) - \sum_{i=1}^{d} \lambda_i^* \nabla^2 c_i(U). \tag{A.164}$$

The Hessian is a $d^2 \times d^2$ matrix, we index it by 4 indices in $[d]$. The entries are summarized below:

$$
\begin{aligned}
[\mathfrak{M}(U)]_{ik,i'k'} &= \frac{\partial}{\partial U_{i'k'}} [\nabla_U \mathcal{L}(U,\lambda)]_{ik}\Big|_{(U,\lambda^*(U))} = \frac{\partial}{\partial U_{i'k'}} [2(\sum_{j:j\neq i} U_{jk}^2 - \lambda) U_{ik}]\Big|_{(U,\lambda^*(U))} \\
&= \begin{cases} 2(\sum_{j:j\neq i} U_{jk}^2 - \lambda_i^*(U)) & \text{if } k = k', i = i' \\ 4U_{i'k}U_{ik} & \text{if } k = k', i \neq i' \\ 0 & \text{if } k \neq k' \end{cases} \\
&= \begin{cases} 2\psi_{ik}(U) & \text{if } k = k', i = i' \\ 4U_{i'k}U_{ik} & \text{if } k = k', i \neq i' \\ 0 & \text{if } k \neq k' \end{cases}
\end{aligned} \tag{A.165}
$$

Similar to the previous case, it is easy to bound the function value and derivatives of the function and the constraints.

Lemma A.3.5. *The objective function (3.9) and p-th order derivative are all bounded by poly(d) for $p = 1, 2, 3$. Each constraint's p-th order derivative is bounded by 2, for $p = 1, 2, 3$.*

Therefore the function satisfy all the smoothness condition we need. Finally we show the gradient and Hessian of Lagrangian satisfy the $(\lambda, \gamma, \varepsilon, \delta)$-strict saddle property. Again we did not try to optimize the dependency with respect to d.

Theorem A.3.6. *Optimization problem (3.9) has exactly $2^d \cdot d!$ local minimum that corresponds to permutation and sign flips of a_i's. Further, it satisfy $(\lambda, \gamma, \varepsilon, \delta)$-strict saddle for $\lambda = 1$ and $\gamma, \varepsilon, \delta = 1/poly(d)$.*

Again, in order to prove this theorem, we follow the same strategy: we consider the transformed version Eq.A.157. and first prove the following lemmas for points around saddle point and local

minimum respectively. We choose

$$\varepsilon_0 = (10d)^{-6},\ \ \varepsilon = 2\varepsilon_0^6,\ \ \delta = 2d\varepsilon_0,\ \ \gamma = \varepsilon_0^4/4,\ \ \mathfrak{S}(u) = \{k \mid |u_k| > \varepsilon_0\} \tag{A.166}$$

Where by intuition, $\mathfrak{S}(u)$ is the set of coordinates whose value is relative large.

Lemma A.3.7. *Under the choice of parameters in Eq.(A.166), suppose $\|\chi(U)\| \le \varepsilon$, and there exists $(i, j) \in [d]^2$ so that $\mathfrak{S}(u^{(i)}) \cap \mathfrak{S}(u^{(j)}) \neq \emptyset$. Then, there exists $\hat{v} \in \mathcal{T}(U)$ and $\|\hat{v}\| = 1$, so that $\hat{v}^T \mathfrak{M}(U)\hat{v} \le -\gamma$.*

Proof. Again, since $\|\chi(x)\| \le \varepsilon = 2\varepsilon_0^6$, by Eq.(A.163), we have for each $i \in [d]$, $|[\chi(x)]_{ik}| = 2|U_{ik}\psi_{ik}(U)| \le 2\varepsilon_0^6$. Therefore, have:

$$\forall k \in \mathfrak{S}(u^{(i)}), \qquad |\psi_{ik}(U)| \le \varepsilon_0^5 \tag{A.167}$$

Then, we prove this lemma by dividing it into three cases. Note in order to prove that there exists $\hat{v} \in \mathcal{T}(U)$ and $\|\hat{v}\| = 1$, so that $\hat{v}^T \mathfrak{M}(U)\hat{v} \le -\gamma$; it suffices to find a vector $v \in \mathcal{T}(U)$ and $\|v\| \le 1$, so that $v^T \mathfrak{M}(U)v \le -\gamma$.

Case 1 : $|\mathfrak{S}(u^{(i)})| \ge 2$, $|\mathfrak{S}(u^{(j)})| \ge 2$, and $|\mathfrak{S}(u^{(i)}) \cap \mathfrak{S}(u^{(j)})| \ge 2$.

WLOG, assume $\{1, 2\} \in \mathfrak{S}(u^{(i)}) \cap \mathfrak{S}(u^{(j)})$, choose v to be $v_{i1} = \frac{U_{i2}}{4}$, $v_{i2} = -\frac{U_{i1}}{4}$, $v_{j1} = \frac{U_{j2}}{4}$ and $v_{j2} = -\frac{U_{j1}}{4}$. All other entries of v are zero. Clearly $v \in \mathcal{T}(U)$, and $\|v\| \le 1$. On the other hand, we know $\mathfrak{M}(U)$ restricted to these 4 coordinates $(i1, i2, j1, j2)$ is

$$\begin{pmatrix} 2\psi_{i1}(U) & 0 & 4U_{i1}U_{j1} & 0 \\ 0 & 2\psi_{i2}(U) & 0 & 4U_{i2}U_{j2} \\ 4U_{i1}U_{j1} & 0 & 2\psi_{j1}(U) & 0 \\ 0 & 4U_{i2}U_{j2} & 0 & 2\psi_{j2}(U) \end{pmatrix} \tag{A.168}$$

By Eq.(A.167), we know all diagonal entries are $\leq 2\varepsilon_0^5$.

If $U_{i1}U_{j1}U_{i2}U_{j2}$ is negative, we have the quadratic form:

$$\begin{aligned} v^T\mathfrak{M}(U)v =& U_{i1}U_{j1}U_{i2}U_{j2} + \frac{1}{8}[U_{i2}^2\psi_{i1}(U) + U_{i1}^2\psi_{i2}(U) + U_{j2}^2\psi_{j1}(U) + U_{j1}^2\psi_{j2}(U)] \\ \leq& -\varepsilon_0^4 + \varepsilon_0^5 \leq -\frac{1}{4}\varepsilon_0^4 = -\gamma \end{aligned} \tag{A.169}$$

If $U_{i1}U_{j1}U_{i2}U_{j2}$ is positive we just swap the sign of the first two coordinates $v_{i1} = -\frac{U_{i2}}{2}$, $v_{i2} = \frac{U_{i1}}{2}$ and the above argument would still holds.

Case 2 : $|\mathfrak{S}(u^{(i)})| \geq 2$, $|\mathfrak{S}(u^{(j)})| \geq 2$, and $|\mathfrak{S}(u^{(i)}) \cap \mathfrak{S}(u^{(j)})| = 1$.

WLOG, assume $\{1,2\} \in \mathfrak{S}(u^{(i)})$ and $\{1,3\} \in \mathfrak{S}(u^{(j)})$, choose v to be $v_{i1} = \frac{U_{i2}}{4}$, $v_{i2} = -\frac{U_{i1}}{4}$, $v_{j1} = \frac{U_{j3}}{4}$ and $v_{j3} = -\frac{U_{j1}}{4}$. All other entries of v are zero. Clearly $v \in \mathcal{T}(U)$ and $\|v\| \leq 1$. On the other hand, we know $\mathfrak{M}(U)$ restricted to these 4 coordinates $(i1, i2, j1, j3)$ is

$$\begin{pmatrix} 2\psi_{i1}(U) & 0 & 4U_{i1}U_{j1} & 0 \\ 0 & 2\psi_{i2}(U) & 0 & 0 \\ 4U_{i1}U_{j1} & 0 & 2\psi_{j1}(U) & 0 \\ 0 & 0 & 0 & 2\psi_{j3}(U) \end{pmatrix} \tag{A.170}$$

By Eq.(A.167), we know all diagonal entries are $\leq 2\varepsilon_0^5$. If $U_{i1}U_{j1}U_{i2}U_{j3}$ is negative, we have the quadratic form:

$$\begin{aligned} v^T\mathfrak{M}(U)v =& \frac{1}{2}U_{i1}U_{j1}U_{i2}U_{j3} + \frac{1}{8}[U_{i2}^2\psi_{i1}(U) + U_{i1}^2\psi_{i2}(U) + U_{j3}^2\psi_{j1}(U) + U_{j1}^2\psi_{j3}(U)] \\ \leq& -\frac{1}{2}\varepsilon_0^4 + \varepsilon_0^5 \leq -\frac{1}{4}\varepsilon_0^4 = -\gamma \end{aligned} \tag{A.171}$$

If $U_{i1}U_{j1}U_{i2}U_{j3}$ is positive we just swap the sign of the first two coordinates $v_{i1} = -\frac{U_{i2}}{2}$, $v_{i2} = \frac{U_{i1}}{2}$ and the above argument would still holds.

Case 3 : Either $|\mathfrak{S}(u^{(i)})| = 1$ or $|\mathfrak{S}(u^{(j)})| = 1$.

WLOG, suppose $|\mathfrak{S}(u^{(i)})| = 1$, and $\{1\} = \mathfrak{S}(u^{(i)})$, we know:

$$|(u_1^{(i)})^2 - 1| \leq (d-1)\varepsilon_0^2 \tag{A.172}$$

On the other hand, since $\mathfrak{S}(u^{(i)}) \cap \mathfrak{S}(u^{(j)}) \neq \emptyset$, we have $\mathfrak{S}(u^{(i)}) \cap \mathfrak{S}(u^{(j)}) = \{1\}$, and thus:

$$|\psi_{j1}(U)| = |\sum_{i':i'\neq j} U_{i'1}^2 - \sum_{i':i'\neq j} h(u^{(i')}, u^{(j)})| \leq \varepsilon_0^5 \tag{A.173}$$

Therefore, we have:

$$\sum_{i':i'\neq j} h(u^{(i')}, u^{(j)}) \geq \sum_{i':i'\neq j} U_{i'1}^2 - \varepsilon_0^5 \geq U_{i1}^2 - \varepsilon_0^5 \geq 1 - d\varepsilon_0^2 \tag{A.174}$$

and

$$\begin{aligned}\sum_{k=1}^{d} \psi_{jk}(U) =& \sum_{i':i'\neq j} \sum_{k=1}^{d} U_{i'k}^2 - d \sum_{i':i'\neq j} h(u^{(i')}, u^{(j)}) \\ \leq& d - 1 - d(1 - d\varepsilon_0^2) = -1 + d^2\varepsilon_0^2\end{aligned} \tag{A.175}$$

Thus, we know, there must exist some $k' \in [d]$, so that $\psi_{jk'}(U) \leq -\frac{1}{d} + d\varepsilon_0^2$. This means we have "large" negative entry on the diagonal of $\mathfrak{M}$. Since $|\psi_{j1}(U)| \leq \varepsilon_0^5$, we know $k' \neq 1$. WLOG, suppose $k' = 2$, we have $|\psi_{j2}(U)| > \varepsilon_0^5$, thus $|U_{j2}| \leq \varepsilon_0$.

Choose v to be $v_{j1} = \frac{U_{j2}}{2}$, $v_{j2} = -\frac{U_{j1}}{2}$. All other entries of v are zero. Clearly $v \in \mathcal{T}(U)$ and $\|v\| \leq 1$. On the other hand, we know $\mathfrak{M}(U)$ restricted to these 2 coordinates $(j1, j2)$ is

$$\begin{pmatrix} 2\psi_{j1}(U) & 0 \\ 0 & 2\psi_{j2}(U) \end{pmatrix} \tag{A.176}$$

We know $|U_{j1}| > \varepsilon_0$, $|U_{j2}| \leq \varepsilon_0$, $|\psi_{j1}(U)| \leq \varepsilon_0^5$, and $\psi_{j2}(U) \leq -\frac{1}{d} + d\varepsilon_0^2$. Thus:

$$v^T \mathfrak{M}(U) v = \frac{1}{2}\psi_{j1}(U)U_{j2}^2 + \frac{1}{2}\psi_{j2}(U)U_{j1}^2$$

$$\leq \varepsilon_0^7 - (\frac{1}{d} - d\varepsilon_0^2)\varepsilon_0^2 \leq -\frac{1}{2d}\varepsilon_0^2 \leq -\gamma \tag{A.177}$$

Since by our choice of v, we have $\|v\| \leq 1$, we can choose $\hat{v} = v/\|v\|$, and immediately have $\hat{v} \in \mathcal{T}(U)$ and $\|\hat{v}\| = 1$, and $\hat{v}^T \mathfrak{M}(U)\hat{v} \leq -\gamma$. □

Lemma A.3.8. *Under the choice of parameters in Eq.(A.166), suppose $\|\chi(U)\| \leq \varepsilon$, and for any $(i, j) \in [d]^2$ we have $\mathfrak{S}(u^{(i)}) \cap \mathfrak{S}(u^{(j)}) = \emptyset$. Then, there is a local minimum $U^\star$ such that $\|U - U^\star\| \leq \delta$, and for all U' in the 2δ neighborhood of $U^\star$, we have $\hat{v}^T \mathfrak{M}(U')\hat{v} \geq 1$ for all $\hat{v} \in \mathcal{T}(U')$, $\|\hat{v}\| = 1$*

Proof. WLOG, we assume $\mathfrak{S}(u^{(i)}) = \{i\}$ for $i = 1, \cdots, d$. Then, we immediately have:

$$|u_j^{(i)}| \leq \varepsilon_0, \qquad |(u_i^{(i)})^2 - 1| \leq (d-1)\varepsilon_0^2, \qquad \forall (i, j) \in [d]^2, j \neq i \tag{A.178}$$

Then $u_i^{(i)} \geq \sqrt{1 - d\varepsilon_0^2}$ or $u_i^{(i)} \leq -\sqrt{1 - d\varepsilon_0^2}$. Which means $u_i^{(i)}$ is either close to 1 or close to -1. By symmetry, we know WLOG, we can assume the case $u_i^{(i)} \geq \sqrt{1 - d\varepsilon_0^2}$ for all $i \in [d]$.

Let $V \in \mathbb{R}^{d^2}$ be the concatenation of $\{e_1, e_2, \cdots, e_d\}$, then we have:

$$\|U - V\|^2 = \sum_{i=1}^{d} \|u^{(i)} - e_i\|^2 \leq 2d^2\varepsilon_0^2 \leq \delta^2 \tag{A.179}$$

Next, we show V is a local minimum. According to Eq.A.165, we know $\mathfrak{M}(V)$ is a diagonal matrix with d^2 entries:

$$[\mathfrak{M}(V)]_{ik,ik} = 2\psi_{ik}(V) = 2\sum_{j:j\neq i}[V_{jk}^2 - \sum_{l=1}^{d} V_{il}^2 V_{jl}^2] = \begin{cases} 2 & \text{if } i \neq k \\ 0 & \text{if } i = k \end{cases} \tag{A.180}$$

We know the unit vector in the direction that corresponds to $[\mathfrak{M}(V)]_{ii,ii}$ is not in the tangent space $\mathcal{T}(V)$ for all $i \in [d]$. Therefore, for any $v \in \mathcal{T}(V)$, we have

$$v^T \mathfrak{M}(e_1)v \geq 2\|v\|^2 > 0 \qquad \text{for all } v \in \mathcal{T}(V), v \neq 0 \tag{A.181}$$

Which by Theorem A.2.6 means V is a local minimum.

Finally, denote $\mathcal{T}_V = \mathcal{T}(V)$ be the tangent space of constraint manifold at V. We know for all U' in the 2δ neighborhood of V, and for all $\hat{v} \in \mathcal{T}(x')$, $\|\hat{v}\| = 1$:

$$\begin{aligned}\hat{v}^T\mathfrak{M}(U')\hat{v} \geq& \hat{v}^T\mathfrak{M}(V)\hat{v} - |\hat{v}^T\mathfrak{M}(V)\hat{v} - \hat{v}^T\mathfrak{M}(U')\hat{v}| \\ =& 2\|P_{\mathcal{T}_V}\hat{v}\|^2 - \|\mathfrak{M}(V) - \mathfrak{M}(U')\|\|\hat{v}\|^2 \\ =& 2 - 2\|P_{\mathcal{T}_V^c}\hat{v}\|^2 - \|\mathfrak{M}(V) - \mathfrak{M}(U')\| \end{aligned} \tag{A.182}$$

By lemma A.2.8, we know $\|P_{\mathcal{T}_V^c}\hat{v}\|^2 \leq \|U' - V\|^2 \leq 4\delta^2$. By Eq.(A.165), we have:

$$\|\mathfrak{M}(V) - \mathfrak{M}(U')\| \leq \|\mathfrak{M}(V) - \mathfrak{M}(U')\| \leq \sum_{(i,j,k)} |[\mathfrak{M}(V)]_{ik,jk} - [\mathfrak{M}(U')]_{ik,jk}| \leq 100d^3\delta \tag{A.183}$$

In conclusion, we have $\hat{v}^T\mathfrak{M}(U')\hat{v} \geq 2 - 8\delta^2 - 100d^3\delta \geq 1$ which finishs the proof. □

Finally, we are ready to prove Theorem A.3.6.

Proof of Theorem A.3.6. Similarly, $(\lambda, \gamma, \varepsilon, \delta)$-strict saddle immediately follows from Lemma A.3.7 and Lemma A.3.8.

The only thing remains to show is that Optimization problem (3.9) has exactly $2^d \cdot d!$ local minimum that corresponds to permutation and sign flips of a_i's. This can be easily proved by the same argument as in the proof of Theorem A.3.2. □

APPENDIX B

APPENDIX FOR TWO LAYER NETWORK CONVERGENCE ANALYSIS

B.1 Flowchart of the proofs

Although the proofs of our theorems are intricate, many lemmas have clear intuition behind the statement. Therefore, we add "*" to these lemmas, so that time constrained readers could feel confident to skip the proofs. We also plot a flowchart of the proofs in Figure B.1 to help the readers spend time wisely.

Since the proofs are long and complicated, we choose to present them in a top-down way. That is, we present the main theorems (Theorem 5.3.1, Theorem 5.3.2, and Theorem 5.3.3) in the main paper, and then present the necessary lemmas in order to prove those main theorems in Section B.2, Section B.3 and Section B.4. Finally, we present the proofs for those lemma in Section B.7, Section B.8 and Section B.9, respectively.

B.2 Compute approximation matrix

The exact form of $-\nabla L(\mathbf{W})_j$ in Lemma 5.2.1 contains variables like $\theta_{i^*,j}, \theta_{i,j}, \sin\theta_{i^*,j}, \sin\theta_{i,j}$, which are hard to deal with. In this section, we compute the approximation of these terms using Taylor series, and show that the approximation loss is minor. While the proofs are technically involved, the claims themselves are not surprising. Hence, we encourage the readers to skip the proofs (Appendix B.7) for the first reading.

Define the j-th column of the approximation matrix $\mathbf{P}$ as follows. See Definition 5.2.2 and

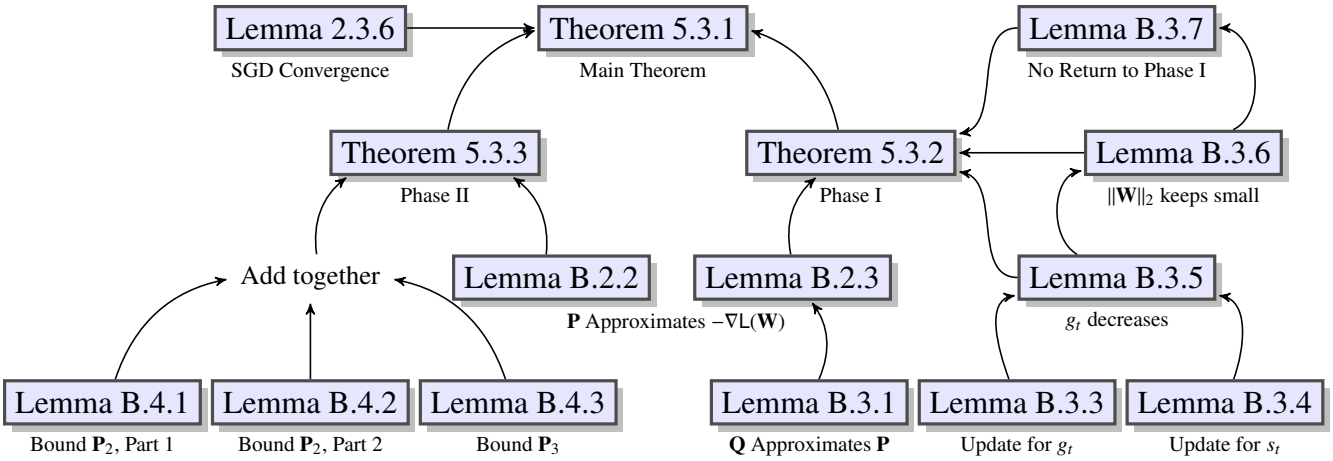


Figure B.1: Flowchart of the proofs

Definition 5.2.3 for $g_j, \mathbf{A}_j$.

$$\begin{aligned}
\mathbf{P}_j &\triangleq \mathbf{P}_{1,j} + \mathbf{P}_{2,j} + \mathbf{P}_{3,j}, \quad \text{where} \\
\mathbf{P}_{1,j} &\triangleq \sum_{i=1}^{d} \frac{\pi}{2}(w_i^* - w_i), \\
\mathbf{P}_{2,j} &\triangleq g_j \overline{e_j + w_j} + \left(\mathbf{I} - \frac{1}{2}\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right)\mathbf{A}_j \overline{e_j + w_j}, \\
\mathbf{P}_{3,j} &\triangleq \left(\frac{\pi}{2} - \theta_{j^*,j}\right)(e_j + w_j^*) - \frac{\pi}{2}(e_j + w_j) + \|e_j + w_j^*\| \sin\theta_{j^*,j} \overline{e_j + w_j}.
\end{aligned}$$

Treat $\mathbf{P}_{1,j}, \mathbf{P}_{2,j}, \mathbf{P}_{3,j}$ as j-th column of matrix $\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$ respectively, we have $\mathbf{P} = \mathbf{P}_1 + \mathbf{P}_2 + \mathbf{P}_3$. Although $\mathbf{P}$ depends on $\mathbf{W}$, we abuse the notation and simply write $\mathbf{P}$.

Claim B.2.1. *$\mathbf{P}_j$ approximates $-\nabla\mathsf{L}(\mathbf{W})_j$ by setting $(\frac{\pi}{2} - \theta_{i,j}) \approx \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle$, $(\frac{\pi}{2} - \theta_{i^*,j}) \approx \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle$, $\sin\theta_{i,j} \approx 1 - \frac{1}{2}\langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2$ and $\sin\theta_{i^*,j} \approx 1 - \frac{1}{2}\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2$.*

Below we show that the approximation loss is negligible in terms of one point convexity and spectral norm.

Lemma* B.2.2. *If $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma \le \frac{1}{100}$, $|\langle \mathbf{P} + \nabla\mathsf{L}(\mathbf{W}), \mathbf{W}^* - \mathbf{W} \rangle| < 0.085\|\mathbf{W}^* - \mathbf{W}\|_F^2$.*

Lemma* B.2.3. *If $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma \le \frac{1}{100}$, $\|\mathbf{P} + \nabla\mathsf{L}(\mathbf{W})\|_2 \le 3.5\gamma^2$.*

B.3 Phase I: the decreasing potential function

As we saw in Theorem 5.3.3, if $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2$ is bounded by a constant $\gamma = \frac{1}{100}$, and the potential function $g \leq 0.1$, $\mathsf{L}(\mathbf{W})$ is 0.03-one point convex, which will give us convergence guarantee according to Lemma 2.3.6. However, g could be larger than 0.1 initially, and as we run SGD, $\|\mathbf{W}\|_2$ might be larger than $\frac{1}{100}$ as well.

In this section, we address both problems by analyzing the dynamics of SGD, thus prove Theorem 5.3.2. The proofs can be found in Appendix B.8. Before proceeding to the interesting stuff, we need a simpler form of $\nabla\mathsf{L}(\mathbf{W})$ to work with, see below.

Lemma B.3.1. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \leq \gamma \leq \frac{1}{100}$, *the negative gradient of* $\mathsf{L}(\mathbf{W})$ *is approximately*

$$\mathbf{Q}(\mathbf{W}) \triangleq \frac{\pi}{2}(\mathbf{W}^* - \mathbf{W})\left(\mathbf{I} + uu^\top\right) + (\mathbf{W}^* - \mathbf{W})^\top - 2\mathrm{Diag}(\mathbf{W}^* - \mathbf{W}) + g\overline{\mathbf{I} + \mathbf{W}}$$

where u *is the all* 1 *vector. The approximation error is* $\|\mathbf{Q}(\mathbf{W}) - [-\nabla\mathsf{L}(\mathbf{W})]\|_2 \leq 61\gamma^2$.

We immediately get the bound of the gradient norm.

Lemma* B.3.2. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \leq \gamma \leq \frac{1}{100}$, $\|\nabla\mathsf{L}(\mathbf{W})\|_2 \leq 6d\gamma$.

Now we are ready to analyze the dynamics. We use subscript t under each variable to denote its value at the step t. For simplicity, let $\mathbf{Q}_t \triangleq \mathbf{Q}(\mathbf{W}_t)$. Define $s_t \triangleq (\mathbf{W}^* - \mathbf{W}_t)u$. We first compute the updating rule for g_t.

Lemma B.3.3. *If* $\|\mathbf{W}_t\|_2, \|\mathbf{W}^*\|_2 \leq \gamma \leq \frac{1}{100}$, $d \geq 100$, $\eta \leq \frac{\gamma^2}{G_2^2}$, *then* $|g_{t+1}| \leq (1 - 0.95\eta d)|g_t| + 86\eta d\gamma^2 + 1.03\eta\sqrt{d}\varepsilon + 4.8\eta\|s_t\|_2\gamma\sqrt{d}$.

The bound contains $\|s_t\|_2$ which could be large, so we also need to compute its updating rule:

Lemma B.3.4. *If* $\|\mathbf{W}_t\|_2, \|\mathbf{W}^*\|_2 \leq \gamma \leq \frac{1}{100}$, *then* $\|s_{t+1}\|_2 \leq \left(1 - \eta\frac{(d+1)\pi}{2}\right)\|s_t\|_2 + \eta(6.61\gamma + 1.03|g_t| + \varepsilon)\sqrt{d}$.

Combining the two lemmas, we are ready to show that g_t will shrink, conditioned on that $\|\mathbf{W}_t\|_2$ is bounded by γ.

Lemma B.3.5. *If for every step* $t > 0$, $\|\mathbf{W}_t\|_2, \|\mathbf{W}^*\|_2 \leq \gamma \leq \frac{1}{100}$,$d \geq 100$, $\eta \leq \frac{\gamma^2}{G_2^2}$, $\varepsilon \leq \gamma^2$, *then* $|g_t|$ *will keep decreasing by a factor of* $1 - 0.5\eta d$ *for every step, until* $|g_{t_1}| \leq 197\gamma^2$ *for* $t_1 \leq \frac{1}{16\eta}$.

Fortunately, we also know that $\|\mathbf{W}_t\|_2$ is always bounded by γ during the process described in Lemma B.3.5.

Lemma B.3.6. *There exists a constant* $\gamma > \gamma_0 > 0$ *such that if* $\|\mathbf{W}_0\|_2, \|\mathbf{W}^*\|_2 \leq \gamma_0$, $d \geq 100$, $\eta \leq \frac{\gamma^2}{G_2^2}$, $\varepsilon \leq \gamma^2$, *then in the process of Phase I (Lemma B.3.5), we always have* $\|\mathbf{W}_T\|_2 \leq \gamma \leq \frac{1}{100}$ *for any* $T > 0$.

Now, we are at the state where $|g_t|$ is small, and $\|\mathbf{W}_T\|_2 \leq \gamma$, which means we are in Phase II. The next lemma ensures that we will stay in Phase II forever.

Lemma B.3.7. *There exists a constant* $\gamma_0 > \gamma > 0$ *such that if* $\|\mathbf{W}_0\|_2, \|\mathbf{W}^*\|_2 \leq \gamma_0$, $d \geq 100$, $\eta \leq \frac{\gamma^2}{G_2^2}$, $\varepsilon \leq \gamma^2$, *then after* $|g_{t_1}| \leq 197\gamma^2$, *Phase I ends and Phase II starts. That is, for every* $T > t_1$, $\|\mathbf{W}_T\|_2 \leq \gamma$ *and* $|g_T| \leq 0.1$.

Proof for Theorem 5.3.2. We immediately get Theorem 5.3.2 by combining the above three lemmas. They show that g_t will decrease to a small value in Phase I (Lemma B.3.5), $\|\mathbf{W}_t\|_2$ will keep small during this process (Lemma B.3.6), and they all keep small afterwards (Lemma B.3.7). □

B.4 Phase II: one point convexity

In this section, we prove Theorem 5.3.3. See detailed proofs in Appendix B.9. Using Lemma B.2.2, it suffices to bound

$$\langle \mathbf{P}, \mathbf{W}^* - \mathbf{W} \rangle = \sum_{j=1}^{d} \langle \mathbf{P}_{1,j} + \mathbf{P}_{2,j} + \mathbf{P}_{3,j}, w_j^* - w_j \rangle$$

Here the first term is easy to calculate.

$$\sum_{j=1}^{d} \langle \mathbf{P}_{1,j}, w_j^* - w_j \rangle = \frac{\pi}{2} \left\| \sum_{i=1}^{d} (w_i^* - w_i) \right\|_2^2 \geq 0 \tag{B.1}$$

For notational simplicity, denote

$$x_j \triangleq \left(\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top \right) (w_j^* - w_j),$$

$$\mathbf{X} \triangleq (x_1, \cdots, x_d) \tag{B.2}$$

$$z_j \triangleq \left(\mathbf{I} - \frac{1}{2} \overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top \right) (w_j^* - w_j) \tag{B.3}$$

By Definition of $\mathbf{P}_{2,j}$ and (B.3), we have

$$\sum_{j=1}^{d} \langle \mathbf{P}_{2,j}, w_j^* - w_j \rangle = \sum_{j=1}^{d} \left\langle g_j \overline{e_j + w_j}, w_j^* - w_j \right\rangle + \sum_{j=1}^{d} z_j^\top \mathbf{A}_j \overline{e_j + w_j} \tag{B.4}$$

We bound the above two terms separately below.

Lemma B.4.1. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \leq \gamma \leq \frac{1}{100}$, *then*

$$\sum_{j=1}^{d} z_j^\top \mathbf{A}_j \overline{e_j + w_j} \geq -(1.3 + 8\gamma) \|\mathbf{W}^* - \mathbf{W}\|_F^2 + \|\mathbf{W}^* - \mathbf{W}\|_F \|\mathbf{X}\|_F.$$

Lemma B.4.2. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \leq \gamma \leq \frac{1}{100}$, *then*

$$\sum_{j=1}^{d} \langle g_j \overline{e_j + w_j}, w_j^* - w_j \rangle \geq -\|\mathbf{W}^* - \mathbf{W}\|_F \|\mathbf{X}\|_F - \frac{(1+\gamma) g \|\mathbf{W}^* - \mathbf{W}\|_F^2}{2(1-2\gamma)}$$

It remains to bound $\sum_{j=1}^{d}\langle \mathbf{P}_{3,j}, w_j^* - w_j\rangle$. We have the following lemma.

Lemma B.4.3. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma \le \frac{1}{100}$, $\sum_{j=1}^{d}\langle \mathbf{P}_{3,j}, w_j^* - w_j\rangle \ge \left(\frac{\pi}{2} - 0.021\right)\|\mathbf{W}^* - \mathbf{W}\|_F^2$.

Proof of Theorem 5.3.3. By (B.1), (B.4), Lemma B.4.1, Lemma B.4.2 and Lemma B.4.3, we know

$$\langle \mathbf{P}, \mathbf{W}^* - \mathbf{W}\rangle \ge \left(\frac{\pi}{2} - 1.321 - 8\gamma - \frac{(1+\gamma)g}{2(1-2\gamma)}\right)\|\mathbf{W}^* - \mathbf{W}\|_F^2 > \left(0.169 - \frac{(1+\gamma)g}{2(1-2\gamma)}\right)\|\mathbf{W}^* - \mathbf{W}\|_F^2$$

Using Lemma B.2.2, we get

$$\langle -\nabla \mathsf{L}(\mathbf{W}), \mathbf{W}^* - \mathbf{W}\rangle > \left(0.084 - \frac{(1+\gamma)g}{2(1-2\gamma)}\right)\|\mathbf{W}^* - \mathbf{W}\|_F^2 > 0.03\|\mathbf{W}^* - \mathbf{W}\|_F^2$$

The last inequality holds when $g \le 0.1$. □

B.5 A geometric lemma

In our proof, we need very tight bounds for a few terms. In order to get such bounds, we present a nice and intuitive geometric lemma as follows.

Lemma B.5.1. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma$, *then* $\forall i \in [d]$,

1. $\|\overline{e_i + w_i^*} - \overline{e_i + w_i}\|_2 \le \frac{\|(\mathbf{I} - \overline{e_i+w_i}\cdot\overline{e_i+w_i}^\top)(w_i^* - w_i)\|_2}{\sqrt{1-2\gamma}} \le \frac{\|w_i^* - w_i\|_2}{\sqrt{1-2\gamma}}$
2. $-\frac{\|w_i^* - w_i\|_2^2}{2(1-2\gamma)} \le \langle \overline{e_i + w_i^*} - \overline{e_i + w_i}, \overline{e_i + w_i}\rangle \le 0$
3. *if* $\gamma \le \frac{1}{100}$, $0 \le \theta_{i,i^*} \le 1.001\|w_i^* - w_i\|_2$.

Proof. See Figure B.2. Denote $e_i + w_i^*$ as $\overrightarrow{OC}$, $e_i + w_i$ as $\overrightarrow{OD}$, $\overline{e_i + w_i^*}$ as $\overrightarrow{OA}$, $\overline{e_i + w_i}$ as $\overrightarrow{OB}$. Thus, $\|w_i^* - w_i\|_2 = \|\overrightarrow{DC}\|_2$.

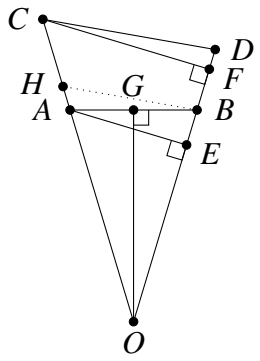


Figure B.2: For Lemma B.5.1

1. Since $\overrightarrow{OD} \perp \overrightarrow{CF}$, we know $\|\overrightarrow{CD}\|_2 \geq \|\overrightarrow{CF}\|_2$. Since $\triangle CFO \sim \triangle AEO$, we know

$$\frac{\|\overrightarrow{CD}\|_2}{\|\overrightarrow{AE}\|_2} \geq \frac{\|\overrightarrow{CF}\|_2}{\|\overrightarrow{AE}\|_2} = \frac{\|\overrightarrow{OC}\|_2}{\|\overrightarrow{OA}\|_2} = \|e_i + w_i^*\|_2 \geq 1 - \gamma \tag{B.5}$$

The last inequality holds as $\|\mathbf{W}^*\|_2 \leq \gamma$.

Notice that $\|\overrightarrow{OA}\|_2 = \|\overrightarrow{OB}\|_2 = 1$, we know $\triangle ABO$ is a isosceles triangle. Thus, $\|\overrightarrow{AG}\|_2 = \|\overrightarrow{GB}\|_2$. Notice that $\triangle ABE \sim \triangle BGO$, we have

$$\frac{\|\overrightarrow{AE}\|_2}{\|\overrightarrow{AB}\|_2} = \frac{\|\overrightarrow{OG}\|_2}{\|\overrightarrow{OB}\|_2} = \frac{\sqrt{1 - \|\overrightarrow{GB}\|_2^2}}{1} \tag{B.6}$$

WLOG, assume $\|\overrightarrow{OC}\|_2 \geq \|\overrightarrow{OD}\|_2$, as shown in the figure. We draw $\overrightarrow{HB} \parallel \overrightarrow{CD}$, and we know $\|\overrightarrow{OH}\|_2 \geq \|\overrightarrow{OB}\|_2 = \|\overrightarrow{OA}\|_2$. Since $\triangle CDO \sim \triangle HBO$, we have

$$\frac{\|\overrightarrow{CD}\|_2}{\|\overrightarrow{HB}\|_2} = \frac{\|\overrightarrow{OD}\|_2}{\|\overrightarrow{OB}\|_2} = \|\overrightarrow{OD}\|_2 \geq 1 - \gamma$$

So $\|\overrightarrow{CD}\|_2 \geq (1-\gamma)\|\overrightarrow{HB}\|_2$. On the other hand, $\angle BAO < \frac{\pi}{2}$, and A is between H and O, so $\angle BAH > \frac{\pi}{2}$, which means $\|\overrightarrow{HB}\|_2 \geq \|\overrightarrow{AB}\|_2 = 2\|\overrightarrow{GB}\|_2$. Thus, $\|\overrightarrow{GB}\|_2 \leq \frac{\|\overrightarrow{HB}\|_2}{2} \leq \frac{\|\overrightarrow{CD}\|_2}{2(1-\gamma)}$.

Substitute it into (B.6), we get

$$\frac{\|\overrightarrow{AE}\|_2}{\|\overrightarrow{AB}\|_2} \geq \sqrt{1 - \frac{\|\overrightarrow{CD}\|_2^2}{4(1-\gamma)^2}} \geq \sqrt{1 - \left(\frac{\gamma}{1-\gamma}\right)^2}$$

The last inequality holds since $\|\overrightarrow{CD}\|_2 = \|w_i^* - w_i\|_2 \le 2\gamma$.

Substitute this inequality into (B.5), we get

$$\begin{aligned}
&\|\overline{e_i + w_i^*} - \overline{e_i + w_i}\|_2 = \|\overrightarrow{AB}\|_2 \\
\le &\frac{\|\overrightarrow{AE}\|_2}{\sqrt{1 - \left(\frac{\gamma}{1-\gamma}\right)^2}} \le \frac{\|\overrightarrow{CF}\|_2}{(1-\gamma)\sqrt{1 - \left(\frac{\gamma}{1-\gamma}\right)^2}} \qquad \text{(B.7)} \\
\le &\frac{\|\overrightarrow{CD}\|_2}{(1-\gamma)\sqrt{1 - \left(\frac{\gamma}{1-\gamma}\right)^2}} = \frac{\|w_i^* - w_i\|_2}{\sqrt{1-2\gamma}} \qquad \text{(B.8)}
\end{aligned}$$

Notice that $\overline{e_i + w_i}^\top (w_i^* - w_i) = -\|\overrightarrow{DF}\|_2$, so $\overline{e_i + w_i} \cdot \overline{e_i + w_i}^\top (w_i^* - w_i) = \overrightarrow{DF}$. That means,

$$\|(\mathbf{I} - \overline{e_i + w_i} \cdot \overline{e_i + w_i}^\top)(w_i^* - w_i)\|_2 = \|\overrightarrow{DC} - \overrightarrow{DF}\|_2 = \|\overrightarrow{CF}\|_2$$

The lemma follows by (B.7) and (B.8).

2. By Figure B.2, we know $|\langle \overline{e_i + w_i^*} - \overline{e_i + w_i}, \overline{e_i + w_i} \rangle| = \|\overrightarrow{BE}\|_2$. Since $\triangle ABE \sim \triangle GBO$, we have

$$\frac{\|\overrightarrow{BE}\|_2}{\|\overrightarrow{AB}\|_2} = \frac{\|\overrightarrow{GB}\|_2}{\|\overrightarrow{BO}\|_2} = \frac{\|\overrightarrow{AB}\|_2}{2}$$

Therefore, using (B.8) we get

$$|\langle \overline{e_i + w_i^*} - \overline{e_i + w_i}, \overline{e_i + w_i} \rangle| = \frac{\|\overrightarrow{AB}\|_2^2}{2} \le \frac{\|w_i^* - w_i\|_2^2}{2(1-2\gamma)}$$

Moreover, $\langle \overline{e_i + w_i^*} - \overline{e_i + w_i}, \overline{e_i + w_i} \rangle = \langle \overline{e_i + w_i^*}, \overline{e_i + w_i} \rangle - 1 \le 0$.

3. We know that

$$\begin{aligned}
\theta_{i,i^*} &= 2 \arcsin \|\overrightarrow{AG}\|_2 = 2 \arcsin \frac{\|\overline{e_i + w_i^*} - \overline{e_i + w_i}\|_2}{2} \\
&\le \|\overline{e_i + w_i^*} - \overline{e_i + w_i}\|_2 + \frac{\|\overline{e_i + w_i^*} - \overline{e_i + w_i}\|_2^3}{8}
\end{aligned}$$

The last inequality holds by Taylor's Series for arcsin, and the fact $\|\overline{e_i + w_i^*} - \overline{e_i + w_i}\|_2 = \|\overrightarrow{AB}\|_2 \le \|w_i^* - w_i\|_2 \le 2\gamma \le \frac{1}{50}$. Thus, we have $\theta_{i,i^*} \le 1.001 \|w_i^* - w_i\|_2$. □

B.6 More handy lemmas

Lemma* B.6.1. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma$, *then*

- $\frac{(1-\gamma)^2}{(1+\gamma)^2}\mathbf{I} \preceq \overline{\mathbf{I}+\mathbf{W}}^\top \overline{\mathbf{I}+\mathbf{W}} \preceq \frac{(1+\gamma)^2}{(1-\gamma)^2}\mathbf{I}, \quad \frac{(1-\gamma)^2}{(1+\gamma)^2}\mathbf{I} \preceq \overline{\mathbf{I}+\mathbf{W}^*}^\top \overline{\mathbf{I}+\mathbf{W}^*} \preceq \frac{(1+\gamma)^2}{(1-\gamma)^2}\mathbf{I},$
- $(1-\gamma)^2\mathbf{I} \preceq (\mathbf{I}+\mathbf{W})^\top(\mathbf{I}+\mathbf{W}) \preceq (1+\gamma)^2\mathbf{I}, \quad (1-\gamma)^2\mathbf{I} \preceq (\mathbf{I}+\mathbf{W}^*)^\top(\mathbf{I}+\mathbf{W}^*) \preceq (1+\gamma)^2\mathbf{I}.$

Therefore, the singular value of $\overline{\mathbf{I}+\mathbf{W}}$ *is at most* $\frac{1+\gamma}{1-\gamma}$ *and at least* $\frac{1-\gamma}{1+\gamma}$. *The singular value of* $\mathbf{I}+\mathbf{W}$ *is at most* $1+\gamma$ *and at least* $1-\gamma$. *The same claims hold for* $\overline{\mathbf{I}+\mathbf{W}^*}$, $\mathbf{I}+\mathbf{W}^*$ *respectively.*

Proof. Since $\|\mathbf{W}\|_2 \le \gamma$, we have $1-\gamma \le \|\mathbf{I}+\mathbf{W}\|_2 \le 1+\gamma$, and $1-\gamma \le \|e_i+w_i\|_2 \le 1+\gamma$. Therefore, $\overline{\mathbf{I}+\mathbf{W}} = \mathbf{\Sigma}(\mathbf{I}+\mathbf{W})$ where $\mathbf{\Sigma}$ is a diagonal matrix whose entries are within $[\frac{1}{1+\gamma}, \frac{1}{1-\gamma}]$. Putting into $\overline{\mathbf{I}+\mathbf{W}}^\top \overline{\mathbf{I}+\mathbf{W}}$, we have

$$\overline{\mathbf{I}+\mathbf{W}}^\top \overline{\mathbf{I}+\mathbf{W}} = (\mathbf{I}+\mathbf{W})^\top \mathbf{\Sigma}^2 (\mathbf{I}+\mathbf{W}) \preceq \frac{1}{(1-\gamma)^2}(\mathbf{I}+\mathbf{W})^\top(\mathbf{I}+\mathbf{W}) \preceq \frac{(1+\gamma)^2}{(1-\gamma)^2}\mathbf{I}$$

Similarly we can show $\overline{\mathbf{I}+\mathbf{W}}^\top \overline{\mathbf{I}+\mathbf{W}} \succeq \frac{(1-\gamma)^2}{(1+\gamma)^2}\mathbf{I}$. Thus we know the singular value of $\overline{\mathbf{I}+\mathbf{W}}$ is at most $\frac{1+\gamma}{1-\gamma}$ and at least $\frac{1-\gamma}{1+\gamma}$. The same proof works for $\mathbf{I}+\mathbf{W}$, $\overline{\mathbf{I}+\mathbf{W}^*}$ and $\mathbf{I}+\mathbf{W}^*$. □

Lemma* B.6.2. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma \le \frac{1}{100}$, *we have*

$$|\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle| \le 2.1\gamma, \quad |\langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle| \le 2.1\gamma$$

Proof. We know

$$|\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle| = \frac{|\langle e_i + w_i^*, e_j + w_j \rangle|}{\|e_i + w_i^*\|_2 \|e_j + w_j\|_2} \le \frac{|\langle e_i + w_i^*, e_j + w_j \rangle|}{(1-\gamma)^2} = \frac{|w_{i,j}^*| + |w_{i,j}| + |\langle w_i, w_j \rangle|}{(1-\gamma)^2} \le \frac{(2+\gamma)\gamma}{(1-\gamma)^2} \le 2.1\gamma$$

where the last inequality holds since $\gamma \le \frac{1}{100}$. The same analysis works for $\langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle$. □

Lemma* B.6.3 (Triangle inequality between $e_i + w_i, e_i + w_i^*, w_i^* - w_i$)**.** $|\|e_i + w_i\|_2 - \|e_i + w_i^*\|_2| \le \|w_i^* - w_i\|_2$.

Lemma* B.6.4. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma$, $|g| \le 2d\gamma$.

Proof. By definition and Lemma B.6.3, we know $|g| = \sum_{i=1}^d (\|e_i+w_i^*\|_2 - \|e_i+w_i\|_2) \le \sum_{i=1}^d \|w_i^* - w_i\|_2 \le 2d\gamma$. □

Lemma* B.6.5. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma$, $|\langle \overline{e_i + w_i^*} - \overline{e_i + w_i}, \overline{e_j + w_j} \rangle| \le \frac{\|w_i^* - w_i\|_2}{\sqrt{1-2\gamma}}$.

Proof. By Cauchy Schwartz and Lemma B.5.1 term 1. □

Lemma* B.6.6. $|x^k - y^k| \le \frac{k}{2}|x - y|(|x|^{k-1} + |y|^{k-1})$.

Proof. $|x^k - y^k| = \left|(x - y) \sum_{t=1}^{k-1} \frac{x^t y^{k-t-1} + y^t x^{k-t-1}}{2}\right| \le \frac{k}{2}|x-y|(|x|^{k-1} + |y|^{k-1})$, where the last inequality holds since $|x^t y^{k-t-1} + y^t x^{k-t-1}| \le |x|^t |y|^{k-t-1} + |y|^t |x|^{k-t-1} \le |x|^{k-1} + |y|^{k-1}$, by rearrangement inequality. □

Lemma* B.6.7. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma \le \frac{1}{100}$, *for* $k \ge 3$, *we have*

$$\begin{aligned}
&\|\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^k (e_i + w_i^*) - \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^k (e_i + w_i)\|_2 \\
\le& 6(2.2\gamma)^{k-3} \left(\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 + \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 \right) \|w_i^* - w_i\|_2
\end{aligned}$$

Proof.

$$\begin{aligned}
&\|\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^k (e_i + w_i^*) - \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^k (e_i + w_i)\|_2 \\
\le& \|w_i^* - w_i\|_2 |\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^k| + \|(\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^k - \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^k)(e_i + w_i)\|_2 \\
\le& \|w_i^* - w_i\|_2 |\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^k| + (1+\gamma)|\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^k - \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^k| \\
\overset{①}{\le}& \|w_i^* - w_i\|_2 (2.1\gamma)^{k-2} \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \\
&+ \frac{(1+\gamma)k}{2} |\langle \overline{e_i + w_i^*} - \overline{e_i + w_i}, \overline{e_j + w_j} \rangle| (|\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle|^{k-1} + |\langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle|^{k-1})
\end{aligned}$$

$$
\begin{aligned}
&\leq \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \left(\|w_i^* - w_i\|_2 (2.1\gamma)^{k-2} + \frac{(1+\gamma)k(2.1\gamma)^{k-3}}{2} |\langle \overline{e_i + w_i^*} - \overline{e_i + w_i}, \overline{e_j + w_j} \rangle| \right) \\
&\quad + \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 \left(\frac{(1+\gamma)k(2.1\gamma)^{k-3}}{2} |\langle \overline{e_i + w_i^*} - \overline{e_i + w_i}, \overline{e_j + w_j} \rangle| \right) \\
&\overset{②}{\leq} \|w_i^* - w_i\|_2 \left[\left((2.1\gamma)^{k-2} + 0.52k(2.1\gamma)^{k-3} \right) \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 + 0.52k(2.1\gamma)^{k-3} \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 \right] \\
&\overset{③}{\leq} \|w_i^* - w_i\|_2 \left[0.55k(2.1\gamma)^{k-3} \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 + 0.52k(2.1\gamma)^{k-3} \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 \right] \\
&\overset{④}{\leq} 6(2.2\gamma)^{k-3} \left(\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 + \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 \right) \|w_i^* - w_i\|_2
\end{aligned}
$$

where ① uses Lemma B.6.2 and Lemma B.6.6, ② uses Lemma B.6.5, ③ holds as $\gamma \leq \frac{1}{100}$, and ④ holds since $0.55k(2.1)^{k-3} \leq 6(2.2)^{k-3}$ for $k \geq 3$. □

Lemma* B.6.8. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \leq \gamma \leq \frac{1}{100}$, *for* $k \geq 2$,

$$
\begin{aligned}
&\left| \|e_i + w_i\|_2 \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^{2k} - \|e_i + w_i^*\|_2 \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^{2k} \right| \\
&\leq 8(2.2\gamma)^{2k-3} \left(\langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 + \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \right) \|w_i^* - w_i\|_2
\end{aligned}
$$

Proof.

$$
\begin{aligned}
&\left| \|e_i + w_i\|_2 \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^{2k} - \|e_i + w_i^*\|_2 \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^{2k} \right| \\
&\leq \|e_i + w_i\|_2 \left| \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^{2k} - \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^{2k} \right| + \left| \|e_i + w_i\|_2 - \|e_i + w_i^*\|_2 \right| \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^{2k} \\
&\overset{①}{\leq} \|e_i + w_i\|_2 \left| \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^{2k} - \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^{2k} \right| + \|w_i^* - w_i\|_2 (2.1\gamma)^{2k-2} \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \\
&\overset{②}{\leq} (1+\gamma)k |\langle \overline{e_i + w_i} - \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle| \left(|\langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle|^{2k-1} + |\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle|^{2k-1} \right) \\
&\quad + \|w_i^* - w_i\|_2 (2.1\gamma)^{2k-2} \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \\
&\overset{③}{\leq} \left[\frac{(1+\gamma)k(2.1\gamma)^{2k-3}}{\sqrt{1-2\gamma}} \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 + \left(\frac{(1+\gamma)k(2.1\gamma)^{2k-3}}{\sqrt{1-2\gamma}} + (2.1\gamma)^{2k-2} \right) \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \right] \|w_i^* - w_i\|_2 \\
&\overset{④}{\leq} 1.05k(2.1\gamma)^{2k-3} \left(\langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 + \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \right) \|w_i^* - w_i\|_2 \\
&\overset{⑤}{\leq} 8(2.2\gamma)^{2k-3} \left(\langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 + \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \right) \|w_i^* - w_i\|_2
\end{aligned}
$$

where ① uses Lemma B.6.2 and Lemma B.6.3, ② uses Lemma B.6.6, ③ uses Lemma B.6.5, ④ holds as $\gamma \leq \frac{1}{100}$, and ⑥ holds as $1.05k(2.1)^{2k-3} \leq 8(2.2)^{2k-3}$ for $k \geq 2$. □

Lemma* B.6.9. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma$*, for fixed* $j \in [d]$*,*

$$\sum_{i\neq j}\langle\overline{e_i + w_i}, \overline{e_j + w_j}\rangle^2 \le \frac{4\gamma}{(1-\gamma)^2}, \qquad \sum_{i\neq j}\langle\overline{e_i + w_i^*}, \overline{e_j + w_j}\rangle^2 \le \frac{4\gamma(1+\gamma)}{1-2\gamma}.$$

Similarly, for fixed $i \in [d]$*,*

$$\sum_{j\neq i}\langle\overline{e_i + w_i}, \overline{e_j + w_j}\rangle^2 \le \frac{4\gamma}{(1-\gamma)^2}, \qquad \sum_{j\neq i}\langle\overline{e_i + w_i^*}, \overline{e_j + w_j}\rangle^2 \le \frac{4\gamma(1+\gamma)}{1-2\gamma}.$$

Proof. By matrix multiplication,

$$\sum_{i=1}^{d}\langle\overline{e_i + w_i^*}, \overline{e_j + w_j}\rangle^2 = \sum_{i=1}^{d}\overline{e_j + w_j}^\top\overline{e_i + w_i^*}\cdot\overline{e_i + w_i^*}^\top\overline{e_j + w_j} = \overline{e_j + w_j}^\top\overline{\mathbf{I} + \mathbf{W}^*}\cdot\overline{\mathbf{I} + \mathbf{W}^*}^\top\overline{e_j + w_j}$$

By Lemma B.6.1, we know $\overline{\mathbf{I} + \mathbf{W}^*}\cdot\overline{\mathbf{I} + \mathbf{W}^*}^\top \le \frac{(1+\gamma)^2}{(1-\gamma)^2}\mathbf{I}$. That means, $\sum_{i=1}^{d}\langle\overline{e_i + w_i^*}, \overline{e_j + w_j}\rangle^2 \le \frac{(1+\gamma)^2}{(1-\gamma)^2}$. On the other hand, by Lemma B.5.1 term 2, $\langle\overline{e_j + w_j^*}, \overline{e_j + w_j}\rangle^2 = (1-\langle\overline{e_j + w_j^*} - \overline{e_j + w_j}, \overline{e_j + w_j}\rangle)^2 \ge 1 - \frac{\|w_i^* - w_i\|_2^2}{1-2\gamma}$.

Therefore, we know

$$\sum_{i\neq j}\langle\overline{e_i + w_i^*}, \overline{e_j + w_j}\rangle^2 \le \frac{(1+\gamma)^2}{(1-\gamma)^2} - 1 + \frac{\|w_i^* - w_i\|_2^2}{1-2\gamma} = \frac{4\gamma}{(1-\gamma)^2} + \frac{\|w_i^* - w_i\|_2^2}{1-2\gamma} \le \frac{4\gamma(1+\gamma)}{1-2\gamma}$$

Using the same analysis, we get $\sum_{i\neq j}\langle\overline{e_i + w_i}, \overline{e_j + w_j}\rangle^2 \le \frac{(1+\gamma)^2}{(1-\gamma)^2} - 1 = \frac{4\gamma}{(1-\gamma)^2}$. The analysis for fixed i is similar. □

Lemma* B.6.10. *For any matrix* $\mathbf{A}$*, we have* $\|\text{Diag}(\mathbf{A})\|_2 \le \|\mathbf{A}\|_2$ *and* $\|\text{Off-Diag}(\mathbf{A})\|_2 \le 2\|\mathbf{A}\|_2$.

Proof. By definition, we know $\|\text{Diag}(\mathbf{A})\|_2 = \max_{i\in[d]} e_i^\top\mathbf{A}e_i \le \max_{v\in\mathbb{R}^d} v^\top\mathbf{A}v = \|\mathbf{A}\|_2$, and $\|\text{Off-Diag}(\mathbf{A})\|_2 \le \|\mathbf{A}\|_2 + \|\text{Diag}(\mathbf{A})\|_2 \le 2\|\mathbf{A}\|_2$. □

Lemma* B.6.11. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma$*,* $\|\mathbf{A}\|_2 \le \frac{2\gamma(\gamma^2+3)}{1-\gamma^2}$.

Proof. By Lemma B.6.1, we have

$$\|\mathbf{A}\|_2 = \|(\mathbf{I} + \mathbf{W}^*)\overline{\mathbf{I} + \mathbf{W}^*}^\top - (\mathbf{I} + \mathbf{W})\overline{\mathbf{I} + \mathbf{W}}^\top\|_2 \le \frac{(1+\gamma)^2}{1-\gamma} - \frac{(1-\gamma)^2}{1+\gamma} = \frac{2\gamma(\gamma^2+3)}{1-\gamma^2}.$$ □

Lemma* B.6.12. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma \le \frac{1}{100}$, $|\overline{e_j + w_j}^\top \mathbf{A} \overline{e_j + w_j} - e_j^\top \mathbf{A} e_j| \le 5\gamma^2$.

Proof.

$$|\overline{e_j + w_j}^\top \mathbf{A} \overline{e_j + w_j} - e_j^\top \mathbf{A} e_j| \le |\overline{e_j + w_j}^\top \mathbf{A}(\overline{e_j + w_j} - e_j)| + |(\overline{e_j + w_j} - e_j)^\top \mathbf{A} e_j| \overset{①}{\le} \frac{4\gamma^2(\gamma^2+3)}{1-\gamma^2} \overset{②}{<} 5\gamma^2$$

where ① uses Cauchy Schwartz, Lemma B.6.11 and $\|\overline{e_j + w_j} - e_j\|_2 \le \gamma$, and ② holds as $\gamma \le \frac{1}{100}$. □

Lemma* B.6.13. *For any* $i \in [n]$, $|\|[e_i + w_i^*\|_2 - \|e_i + w_i\|_2] - [w_{i,i}^* - w_{i,i}]| \le 6.07\gamma^2$.

Proof.

$$\begin{aligned}
&\|e_i + w_i\|_2 - \|e_i + w_i^*\|_2 = \langle e_i + w_i, \overline{e_i + w_i}\rangle - \langle e_i + w_i^*, \overline{e_i + w_i^*}\rangle \\
&= \langle e_i + w_i, \overline{e_i + w_i} - \overline{e_i + w_i^*}\rangle + \langle w_i - w_i^*, \overline{e_i + w_i^*}\rangle \\
&= \langle w_i - w_i^*, e_i\rangle + \langle e_i + w_i, \overline{e_i + w_i} - \overline{e_i + w_i^*}\rangle + \langle w_i - w_i^*, \overline{e_i + w_i^*} - e_i\rangle \\
&= w_{i,i} - w_{i,i}^* + \langle e_i + w_i, \overline{e_i + w_i} - \overline{e_i + w_i^*}\rangle + \langle w_i - w_i^*, \overline{e_i + w_i^*} - e_i\rangle
\end{aligned}$$

As a result,

$$\begin{aligned}
&|[\|e_i + w_i\|_2 - \|e_i + w_i^*\|_2] - [w_{i,i} - w_{i,i}^*]| \le |\langle e_i + w_i, \overline{e_i + w_i} - \overline{e_i + w_i^*}\rangle| + |\langle w_i - w_i^*, \overline{e_i + w_i^*} - e_i\rangle| \\
&\overset{①}{\le} \frac{(1+\gamma)2\gamma^2}{1-2\gamma} + 4\gamma^2 \le 6.07\gamma^2
\end{aligned}$$

where ① uses Lemma B.5.1 term 2 and $\|\overline{e_i + w_i^*} - e_i\|_2 \le 2\gamma$, and Cauchy Schwartz. So the claim follows. □

Corollary B.6.14. $|g - \mathrm{Tr}(\mathbf{W}^* - \mathbf{W})| \le 6.07d\gamma^2$.

Lemma* B.6.15. $\overline{\mathbf{I} + \mathbf{W}}$ *is close to* $\mathbf{I}$ *on its diagonals, and close to* $\mathbf{W}$ *on its off-diagonals. More specifically, if* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma \le \frac{1}{100}$,

$$\|\mathrm{Diag}(\overline{\mathbf{I} + \mathbf{W}}) - \mathbf{I}\|_2 \le \frac{\gamma^2}{2(1-\gamma)^2}, \qquad \|\mathrm{Diag}(\overline{\mathbf{I} + \mathbf{W}^*}) - \mathbf{I}\|_2 \le \frac{\gamma^2}{2(1-\gamma)^2}$$

$$\|\text{Off-Diag}(\overline{\mathbf{I}+\mathbf{W}}-\mathbf{W})\|_2 \le \frac{4\gamma^2}{1-\gamma}, \qquad \|\text{Off-Diag}(\overline{\mathbf{I}+\mathbf{W}^*}-\mathbf{W}^*)\|_2 \le \frac{4\gamma^2}{1-\gamma}$$

$$\|\overline{\mathbf{I}+\mathbf{W}}-\mathbf{I}\|_2 \le 2.05\gamma, \qquad \|\overline{\mathbf{I}+\mathbf{W}^*}-\mathbf{I}\|_2 \le 2.05\gamma$$

Proof. For the diagonal terms,

$$\|\text{Diag}(\overline{\mathbf{I}+\mathbf{W}})-\mathbf{I}\|_2 = \max_j |\overline{\mathbf{I}+\mathbf{W}}_{j,j}-1| = \max_j \left|\frac{1+w_{j,j}-\|e_j+w_j\|_2}{\|e_j+w_j\|_2}\right|$$
$$\le \max_j \left|\frac{(1+w_{j,j})^2-\|e_j+w_j\|_2^2}{\|e_j+w_j\|_2}\right|\left|\frac{1}{1+w_{j,j}+\|e_j+w_j\|_2}\right| \le \max_j \frac{\sum_{i\neq j} w_{j,i}^2}{2(1-\gamma)^2} \le \frac{\gamma^2}{2(1-\gamma)^2}$$

For the off-diagonal terms, we know $\overline{\mathbf{I}+\mathbf{W}} = (\mathbf{I}+\mathbf{W})\mathbf{\Sigma}$ for some diagonal matrix $\mathbf{\Sigma}$, so

$$\|\text{Off-Diag}(\overline{\mathbf{I}+\mathbf{W}}-\mathbf{W})\|_2 = \|\text{Off-Diag}((\mathbf{I}+\mathbf{W})\mathbf{\Sigma}-\mathbf{W})\|_2 = \|\text{Off-Diag}((\mathbf{\Sigma}-\mathbf{I})\mathbf{W})\|_2 \overset{①}{\le} 2\|(\mathbf{\Sigma}-\mathbf{I})\mathbf{W}\|_2 \le \frac{4\gamma^2}{1-\gamma}$$

where ① uses Lemma B.6.10. For the difference between $\overline{\mathbf{I}+\mathbf{W}}$ and $\mathbf{I}$, we split $\overline{\mathbf{I}+\mathbf{W}}$ into diagonal and off-diagonal parts:

$$\|\overline{\mathbf{I}+\mathbf{W}}-\mathbf{I}\|_2 = \|\text{Diag}(\overline{\mathbf{I}+\mathbf{W}})+\text{Off-Diag}(\overline{\mathbf{I}+\mathbf{W}})-\mathbf{I}\|_2$$
$$=\|\text{Off-Diag}(\mathbf{W})\|_2 + \frac{\gamma^2}{2(1-\gamma)^2} + \frac{4\gamma^2}{1-\gamma} \overset{①}{\le} 2\|\mathbf{W}\|_2 + \frac{\gamma^2(9-8\gamma)}{2(1-\gamma)^2} \le 2.05\gamma$$

where ① uses Lemma B.6.10. □

Lemma* B.6.16. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma \le \frac{1}{100}$,

$$\|\mathbf{A} - [\mathbf{W}^*-\mathbf{W}+(\mathbf{W}^*-\mathbf{W})^\top - \text{Diag}(\mathbf{W}^*-\mathbf{W})]\|_2 \le 9.2\gamma^2$$

Proof. By definition,

$$\left\|\left[(\mathbf{I}+\mathbf{W}^*)\overline{\mathbf{I}+\mathbf{W}^*}^\top - (\mathbf{I}+\mathbf{W})\overline{\mathbf{I}+\mathbf{W}}^\top\right] - \left[(\mathbf{W}^*-\mathbf{W}) + (\overline{\mathbf{I}+\mathbf{W}^*}^\top - \overline{\mathbf{I}+\mathbf{W}}^\top)\right]\right\|_2$$
$$=\|\mathbf{W}^*(\overline{\mathbf{I}+\mathbf{W}^*}^\top - \mathbf{I}) - \mathbf{W}(\overline{\mathbf{I}+\mathbf{W}}^\top - \mathbf{I})\|_2 \le \|\mathbf{W}^*(\overline{\mathbf{I}+\mathbf{W}^*}^\top - \mathbf{I})\|_2 + \|\mathbf{W}(\overline{\mathbf{I}+\mathbf{W}})^\top - \mathbf{I})\|_2$$
$$\le 2.05\gamma^2 + 2.05\gamma^2 = 4.1\gamma^2$$

where the last inequality uses Lemma B.6.15. Below we further approximate $\overline{\mathbf{I}+\mathbf{W}^*}^\top - \overline{\mathbf{I}+\mathbf{W}}^\top$.

$$
\begin{aligned}
&\left\|\left[\overline{\mathbf{I}+\mathbf{W}^*}^\top - \overline{\mathbf{I}+\mathbf{W}}^\top\right] - [(\mathbf{W}^*-\mathbf{W})^\top - \mathrm{Diag}(\mathbf{W}^*-\mathbf{W})]\right\|_2 \\
=&\left\|\mathrm{Diag}(\overline{\mathbf{I}+\mathbf{W}^*}^\top - \overline{\mathbf{I}+\mathbf{W}}^\top) + \text{Off-Diag}(\overline{\mathbf{I}+\mathbf{W}^*}^\top - \overline{\mathbf{I}+\mathbf{W}}^\top) - [(\mathbf{W}^*-\mathbf{W})^\top - \mathrm{Diag}(\mathbf{W}^*-\mathbf{W})]\right\|_2 \\
\overset{①}{\leq}&\|\text{Off-Diag}(\overline{\mathbf{I}+\mathbf{W}^*}^\top - \overline{\mathbf{I}+\mathbf{W}}^\top) - \text{Off-Diag}(\mathbf{W}^*-\mathbf{W})^\top\|_2 + \frac{\gamma^2}{(1-\gamma)^2} \\
\overset{②}{\leq}&\frac{4\gamma^2}{1-\gamma} + \frac{\gamma^2}{(1-\gamma)^2} \leq 5.1\gamma^2
\end{aligned}
$$

where ① uses Lemma B.6.15, ② uses Lemma B.6.15 Combining everything,

$$\|\mathbf{A} - [\mathbf{W}^* - \mathbf{W} + (\mathbf{W}^*-\mathbf{W})^\top - \mathrm{Diag}(\mathbf{W}^*-\mathbf{W})]\|_2 \leq 9.2\gamma^2 \qquad \square$$

Using Lemma B.6.10, we immediately have the following corollary.

Corollary B.6.17. $\|\mathrm{Diag}(\mathbf{A}) - \mathrm{Diag}(\mathbf{W}^*-\mathbf{W})\|_2 \leq 9.2\gamma^2$.

Lemma* B.6.18. *For $\eta \leq \frac{1}{\pi d}$,*

$$\left\|\mathbf{I} - \eta\left(\frac{\pi}{2}uu^\top + \left(\frac{\pi}{2}+1\right)\mathbf{I}\right)\right\|_2 \leq \left(1 - \eta\left(\frac{\pi}{2}+1\right)\right)$$

Proof. Consider another basis $(e_1', \cdots, e_d')$ where $e_1' = \frac{u}{\|u\|_2}$. For every unit vector $v = (v_1, \cdots, v_d)$ in this new space, we know

$$v^T\left(\mathbf{I} - \eta\left(\frac{\pi}{2}uu^\top + \left(\frac{\pi}{2}+1\right)\mathbf{I}\right)\right)v = \|v\|_2^2 - \eta\left(\frac{\pi}{2}+1\right)\|v\|_2^2 - \frac{\pi\eta d}{2}v_1^2$$

Hence we get

$$0 \leq v^T\left(\mathbf{I} - \eta\left(\frac{\pi}{2}uu^\top + \left(\frac{\pi}{2}+1\right)\mathbf{I}\right)\right)v \leq \left(1 - \eta\left(\frac{\pi}{2}+1\right)\right)\|v\|_2^2$$

By definition of matrix norm, the lemma follows. $\square$

B.7 Proofs for Section B.2

B.7.1 Proof for Claim B.2.1

Comparing with Lemma 5.2.1, we know that for fixed j, $\mathbf{P}_{1,j}$ is already contained in $-\nabla L(\mathbf{W})_j$ as the first term, while $\mathbf{P}_{3,j}$ is simply the summand when $i = j$, ignoring the first term. Below we show how to obtain $\mathbf{P}_{2,j}$ from $i \neq j$ cases. We will bound the approximation error in Lemma B.2.2 and Lemma B.2.3.

$$
\begin{aligned}
&\sum_{i\neq j}\left(\left(\frac{\pi}{2}-\theta_{i^*,j}\right)(e_i+w_i^*)-\left(\frac{\pi}{2}-\theta_{i,j}\right)(e_i+w_i)+\left(\|e_i+w_i^*\|\sin\theta_{i^*,j}-\|e_i+w_i\|\sin\theta_{i,j}\right)\overline{e_j+w_j}\right)\\
\approx&\sum_{i\neq j}\left(\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle(e_i+w_i^*)-\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle(e_i+w_i)\right)\\
&+\sum_{i\neq j}\left(\|e_i+w_i^*\|\left(1-\frac{1}{2}\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle^2\right)-\|e_i+w_i\|\left(1-\frac{1}{2}\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle^2\right)\right)\overline{e_j+w_j}\\
=&\sum_{i\neq j}((e_i+w_i^*)\overline{e_i+w_i^*}^\top-(e_i+w_i)\overline{e_i+w_i}^\top)\overline{e_j+w_j}\\
&+\sum_{i\neq j}\left(\|e_i+w_i^*\|-\|e_i+w_i\|-\frac{1}{2}\overline{e_j+w_j}^\top\overline{e_i+w_i^*}\|e_i+w_i^*\|\overline{e_i+w_i^*}^\top\overline{e_j+w_j}\right.\\
&\left.+\frac{1}{2}\overline{e_j+w_j}^\top\overline{e_i+w_i}\|e_i+w_i\|\overline{e_i+w_i}^\top\overline{e_j+w_j}\right)\overline{e_j+w_j}\\
=&\mathbf{A}_j\overline{e_j+w_j}+\left(\sum_{i\neq j}(\|e_i+w_i^*\|-\|e_i+w_i\|)-\sum_{i\neq j}\frac{1}{2}\overline{e_j+w_j}^\top(e_i+w_i^*)\overline{e_i+w_i^*}^\top\overline{e_j+w_j}\right.\\
&\left.+\sum_{i\neq j}\frac{1}{2}\overline{e_j+w_j}^\top(e_i+w_i)\overline{e_i+w_i}^\top\overline{e_j+w_j}\right)\overline{e_j+w_j}\\
=&\mathbf{A}_j\overline{e_j+w_j}+\left(g_j-\frac{1}{2}\overline{e_j+w_j}^\top\mathbf{A}_j\overline{e_j+w_j}\right)\overline{e_j+w_j}=\mathbf{P}_{2,j}.
\end{aligned}
$$

B.7.2 Proof for Lemma B.2.2

In order to prove this lemma, we bound the approximation loss of $\theta_{i,j}, \theta_{i^*,j}$ in Lemma B.7.1, and the approximation loss of $\sin\theta_{i,j}, \sin\theta_{i^*,j}$ in Lemma B.7.2.

Lemma* B.7.1 (Approximation loss related to $\theta_{i,j}, \theta_{i^*,j}$)**.** *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma \le \frac{1}{100}$,

$$\sum_{j=1}^{d}\sum_{i\neq j}\left|\left\langle(\frac{\pi}{2}-\theta_{i^*,j}-\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle)(e_i+w_i^*)-(\frac{\pi}{2}-\theta_{i,j}-\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle)(e_i+w_i), w_j^*-w_j\right\rangle\right|$$

$$\le 0.083\|\mathbf{W}^*-\mathbf{W}\|_F^2$$

Proof. By definition, $\frac{\pi}{2}-\theta_{i^*,j} = \arcsin\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle$, and $\frac{\pi}{2}-\theta_{i,j} = \arcsin\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle$.

The Taylor series of $\arcsin x$ at $x=0$ is $\sum_{k=0}^{\infty}\frac{(2k)!}{4^k(k!)^2(2k+1)}x^{2k+1}$, where for $k\ge 1$,

$$\frac{(2k)!}{4^k(k!)^2(2k+1)} \le \frac{1}{6} \tag{B.9}$$

Thus,

$$\sum_{j=1}^{d}\sum_{i\neq j}\left|\left\langle(\frac{\pi}{2}-\theta_{i^*,j}-\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle)(e_i+w_i^*)-(\frac{\pi}{2}-\theta_{i,j}-\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle)(e_i+w_i), w_j^*-w_j\right\rangle\right|$$

$$\overset{①}{\le}\sum_{j=1}^{d}\sum_{i\neq j}\sum_{k=1}^{\infty}\frac{1}{6}\left|\left\langle\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle^{2k+1}(e_i+w_i^*)-\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle^{2k+1}(e_i+w_i), w_j^*-w_j\right\rangle\right|$$

$$\overset{②}{\le}\sum_{j=1}^{d}\sum_{i\neq j}\sum_{k=1}^{\infty}\frac{1}{6}\left\|\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle^{2k+1}(e_i+w_i^*)-\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle^{2k+1}(e_i+w_i)\right\|_2\|w_j^*-w_j\|_2$$

$$\overset{③}{\le}\sum_{j=1}^{d}\sum_{i\neq j}\sum_{k=1}^{\infty}(2.2\gamma)^{2k-2}\left(\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle^2+\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle^2\right)\|w_i^*-w_i\|_2\|w_j^*-w_j\|_2$$

$$\overset{④}{\le}\sum_{j=1}^{d}\sum_{i\neq j}1.01\left(\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle^2+\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle^2\right)\|w_i^*-w_i\|_2\|w_j^*-w_j\|_2$$

$$\overset{⑤}{\le}1.01\left(\sum_{j=1}^{d}\sum_{i\neq j}\left(\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle^2+\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle^2\right)\|w_i^*-w_i\|_2^2\right)^{\frac{1}{2}}$$

$$
\begin{aligned}
&\left(\sum_{j=1}^{d}\sum_{i\neq j}\left(\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle^2+\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle^2\right)\|w_j^*-w_j\|_2^2\right)^{\frac{1}{2}}\\
\leq& 1.01\left[\sum_{i=1}^{d}\|w_i^*-w_i\|_2^2\left(\sum_{i\neq j}\left(\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle^2+\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle^2\right)\right)\right]^{\frac{1}{2}}\\
&\left[\sum_{j=1}^{d}\|w_j^*-w_j\|_2^2\left(\sum_{i\neq j}\left(\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle^2+\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle^2\right)\right)\right]^{\frac{1}{2}}\\
\overset{⑥}{\leq}& 1.01\left(\frac{4\gamma}{(1-\gamma)^2}+\frac{4\gamma(1+\gamma)}{1-2\gamma}\right)\|\mathbf{W}^*-\mathbf{W}\|_F^2\overset{⑦}{\leq}0.083\|\mathbf{W}^*-\mathbf{W}\|_F^2
\end{aligned}
$$

where ① is by Taylor series, ② uses Cauchy Schwartz, ③ uses Lemma B.6.7, ④ holds as $\gamma\leq\frac{1}{100}$, ⑤ uses Cauchy Schwartz, ⑥ uses Lemma B.6.9, ⑦ holds as $\gamma\leq\frac{1}{100}$.

□

Lemma* B.7.2 (Approximation loss related to $\sin\theta_{i,j},\sin\theta_{i^*,j}$)**.** *If* $\|\mathbf{W}\|_2,\|\mathbf{W}^*\|_2\leq\gamma\leq\frac{1}{100}$,

$$
\begin{aligned}
\sum_{j=1}^{d}\sum_{i\neq j}\Bigg|\Bigg(&\|e_i+w_i^*\|_2\left(\sin\theta_{i^*,j}-1+\frac{1}{2}\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle^2\right)-\\
&\|e_i+w_i\|_2\left(\sin\theta_{i,j}-1+\frac{1}{2}\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle^2\right)\Bigg)\langle\overline{e_j+w_j},w_j^*-w_j\rangle\Bigg|\leq 0.002\|\mathbf{W}^*-\mathbf{W}\|_F^2
\end{aligned}
$$

Proof. By definition, we know $\theta_{i^*,j}=\arccos\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle$, and $\theta_{i,j}=\arccos\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle$. The Taylor series of $\sin(\arccos x)$ at $x=0$ is $1-\frac{x^2}{2}-\frac{x^4}{8}-\frac{x^6}{16}-\frac{5x^8}{128}-\cdots=\sum_{k=0}^{\infty}c_kx^{2k}$, where $c_k\leq\frac{1}{8}$ for $k\geq 2$. Thus,

$$
\begin{aligned}
&\sum_{j=1}^{d}\sum_{i\neq j}\Bigg|\Bigg(\|e_i+w_i^*\|_2\left(\sin\theta_{i^*,j}-1+\frac{1}{2}\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle^2\right)-\\
&\qquad\|e_i+w_i\|_2\left(\sin\theta_{i,j}-1+\frac{1}{2}\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle^2\right)\Bigg)\langle\overline{e_j+w_j},w_j^*-w_j\rangle\Bigg|\\
\overset{①}{\leq}&\sum_{j=1}^{d}\sum_{i\neq j}\left|\sum_{k=2}^{\infty}\frac{1}{8}\left(\|e_i+w_i\|_2\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle^{2k}-\|e_i+w_i^*\|_2\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle^{2k}\right)\right|\|w_j^*-w_j\|_2\\
\overset{②}{\leq}&\sum_{j=1}^{d}\sum_{i\neq j}\sum_{k=2}^{\infty}(2.2\gamma)^{2k-3}\left(\langle\overline{e_i+w_i},\overline{e_j+w_j}\rangle^2+\langle\overline{e_i+w_i^*},\overline{e_j+w_j}\rangle^2\right)\|w_i^*-w_i\|_2\|w_j^*-w_j\|_2
\end{aligned}
$$

$$
\begin{aligned}
&\overset{③}{\leq} 2.3\gamma \left(\sum_{j=1}^{d} \sum_{i \neq j} \left(\langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 + \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \right) \|w_i^* - w_i\|_2^2 \right)^{\frac{1}{2}} \\
&\quad \left(\sum_{j=1}^{d} \sum_{i \neq j} \left(\langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 + \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \right) \|w_j^* - w_j\|_2^2 \right)^{\frac{1}{2}} \\
&\leq 2.3\gamma \left[\sum_{i=1}^{d} \|w_i^* - w_i\|_2^2 \left(\sum_{j \neq i} \left(\langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 + \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \right) \right) \right]^{\frac{1}{2}} \\
&\quad \left[\sum_{j=1}^{d} \|w_j^* - w_j\|_2^2 \left(\sum_{i \neq j} \left(\langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 + \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \right) \right) \right]^{\frac{1}{2}} \\
&\overset{④}{\leq} 2.3\gamma \left(\frac{4\gamma}{(1-\gamma)^2} + \frac{4\gamma(1+\gamma)}{1-2\gamma} \right) \|\mathbf{W}^* - \mathbf{W}\|_F^2 \overset{⑤}{<} 0.002 \|\mathbf{W}^* - \mathbf{W}\|_F^2
\end{aligned}
$$

where ① is by Taylor series, ② uses Lemma B.6.8 and Cauchy Schwartz, ③ uses Cauchy Schwartz and $\gamma \leq \frac{1}{100}$, ④ uses Lemma B.6.9, and ⑤ holds as $\gamma \leq \frac{1}{100}$. □

Proof for Lemma B.2.2. Combining the results from Lemma B.7.1 and Lemma B.7.2, the lemma follows. □

B.7.3 Proof for Lemma B.2.3

Denote $\mathbf{\Delta} \triangleq \mathbf{P} + \nabla \mathsf{L}(\mathbf{W})$. This lemma is harder to prove than the previous one since we need to bound the spectral norm of a matrix $\mathbf{\Delta}$. First of all, we need to represent $\mathbf{\Delta}$. Again, the difference has two parts: approximation for $\theta_{i,j}, \theta_{i^*,j}$, and $\sin\theta_{i,j}, \sin\theta_{i^*,j}$. Denote the two parts as $\mathbf{\Delta}_1, \mathbf{\Delta}_2$, where $\mathbf{\Delta} = \mathbf{\Delta}_1 + \mathbf{\Delta}_2$. From the proof of Lemma B.7.1, we know the j-th column of the first part is

$$
\mathbf{\Delta}_{1,j} \triangleq \sum_{i \neq j} \sum_{k=1}^{\infty} \frac{(2k)!}{4^k (k!)^2 (2k+1)} \left(\langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^{2k+1} (e_i + w_i^*) - \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^{2k+1} (e_i + w_i) \right)
$$

And the j-th column of the second part is

$$
\mathbf{\Delta}_{2,j} \triangleq \sum_{i \neq j} \sum_{k=2}^{\infty} c_k \left(\|e_i + w_i\|_2 \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^{2k} - \|e_i + w_i^*\|_2 \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^{2k} \right) \overline{e_j + w_j}
$$

Below we bound $\|\mathbf{\Delta}_1\|_2$ in Lemma B.7.3, and bounds $\|\mathbf{\Delta}_2\|_2$ in Lemma B.7.4.

Lemma* B.7.3. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma \le \frac{1}{100}$, $\|\mathbf{\Delta}_1\|_2 \le 3.4\gamma^2$.

Proof. Define $\mathbf{U}, \mathbf{V}$ such that for $i = j, \mathbf{U}_{i,j} = \mathbf{V}_{i,j} = 0$, and for $i \neq j$,

$$\mathbf{U}_{i,j} = \sum_{k=1}^{\infty} \frac{(2k)!}{4^k(k!)^2(2k+1)} \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^{2k+1}, \mathbf{V}_{i,j} = \sum_{k=1}^{\infty} \frac{(2k)!}{4^k(k!)^2(2k+1)} \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^{2k+1}$$

By matrix multiplication,

$$\mathbf{\Delta}_1 = \sum_{i=1}^{d} \left[(\mathbf{I} + \mathbf{W}^*)_{*,i} \mathbf{U}_{i,*} - (\mathbf{I} + \mathbf{W})_{*,i} \mathbf{V}_{i,*} \right] = (\mathbf{I} + \mathbf{W}^*)\mathbf{U} - (\mathbf{I} + \mathbf{W})\mathbf{V} \tag{B.10}$$

So it suffices to bound $\|\mathbf{U}\|_2, \|\mathbf{V}\|_2$. For $i \neq j$,

$$|\mathbf{U}_{i,j}| = \left| \sum_{k=1}^{\infty} \frac{(2k)!}{4^k(k!)^2(2k+1)} \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^{2k+1} \right| \overset{①}{\le} \sum_{k=1}^{\infty} \frac{(2.1\gamma)^{2k-1}}{6} \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \le 0.4\gamma \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2$$

where ① uses Lemma B.6.2 and (B.9). Now, we know

$$\|\mathbf{U}\|_1 \overset{①}{=} \max_j \sum_{i=1}^{d} |\mathbf{U}_{i,j}| \le \max_j \sum_{i \neq j} 0.4\gamma \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \overset{②}{\le} \frac{1.6(1+\gamma)\gamma^2}{1 - 2\gamma} \le 1.65\gamma^2$$

where ① is by definition, ② uses Lemma B.6.9. Similarly,

$$\|\mathbf{U}\|_\infty = \max_i \sum_{j=1}^{d} |\mathbf{U}_{i,j}| \le \max_i \sum_{j \neq i} 0.4\gamma \langle \overline{e_i + w_i^*}, \overline{e_j + w_j} \rangle^2 \le 1.65\gamma^2$$

By Hölder's inequality, we have

$$\|\mathbf{U}\|_2 \le \sqrt{\|\mathbf{U}\|_1 \|\mathbf{U}\|_\infty} \le 1.65\gamma^2$$

Now we do the same analysis for $\mathbf{V}$.

$$\begin{aligned} |\mathbf{V}_{i,j}| &= \left| \sum_{k=1}^{\infty} \frac{(2k)!}{4^k(k!)^2(2k+1)} \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^{2k+1} \right| \\ &\le \sum_{k=1}^{\infty} \frac{(2.1\gamma)^{2k-1}}{6} \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 \le 0.4\gamma \langle \overline{e_i + w_i}, \overline{e_j + w_j} \rangle^2 \end{aligned}$$

Hence, $\|\mathbf{V}\|_1 = \max_j \sum_{i=1}^d |\mathbf{V}_{i,j}| \le \max_j \sum_{i\neq j} 0.4\gamma\langle\overline{e_i + w_i}, \overline{e_j + w_j}\rangle^2 \le 1.65\gamma^2$. Similarly, $\|\mathbf{V}\|_\infty \le 1.65\gamma^2$, and by Hölder's inequality, $\|\mathbf{V}\|_2 \le \sqrt{\|\mathbf{V}\|_1\|\mathbf{V}\|_\infty} \le 1.65\gamma^2$. Using (B.10), we get

$$\|\mathbf{\Delta}_1\|_2 \le \|\mathbf{I} + \mathbf{W}^*\|_2\|\mathbf{U}\|_2 + \|\mathbf{I} + \mathbf{W}\|_2\|\mathbf{V}\|_2 \le 2(1+\gamma)1.65\gamma^2 < 3.4\gamma^2 \qquad \square$$

Lemma* B.7.4. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma \le \frac{1}{100}$, $\|\mathbf{\Delta}_2\|_2 \le 6\gamma^3$.

Proof. By definition, we can write

$$\mathbf{\Delta}_2 = \overline{\mathbf{I} + \mathbf{W}}\mathrm{Diag}\left\{\sum_{i\neq j}\sum_{k=2}^{\infty} c_k\left(\|e_i + w_i\|_2\langle\overline{e_i + w_i}, \overline{e_j + w_j}\rangle^{2k} - \|e_i + w_i^*\|_2\langle\overline{e_i + w_i^*}, \overline{e_j + w_j}\rangle^{2k}\right)\right\}_{j=1}^d$$

So it suffices to bound the norm of the diagonal matrix, which is the maximum of the diagonal entries. For any $j \in [d]$, we have

$$\begin{aligned}
&\left|\sum_{i\neq j}\sum_{k=2}^{\infty} c_k\left(\|e_i + w_i\|_2\langle\overline{e_i + w_i}, \overline{e_j + w_j}\rangle^{2k} - \|e_i + w_i^*\|_2\langle\overline{e_i + w_i^*}, \overline{e_j + w_j}\rangle^{2k}\right)\right| \\
\le &\sum_{i\neq j}\sum_{k=2}^{\infty}\frac{1}{8}\left(\|e_i + w_i\|_2\langle\overline{e_i + w_i}, \overline{e_j + w_j}\rangle^{2k}| + |\|e_i + w_i^*\|_2\langle\overline{e_i + w_i^*}, \overline{e_j + w_j}\rangle^{2k}\right) \\
\overset{①}{\le} &\sum_{i\neq j}\sum_{k=2}^{\infty}\frac{1}{4}(1+\gamma)(2.1\gamma)^{2k-2}\left(\langle\overline{e_i + w_i}, \overline{e_j + w_j}\rangle^2 + \langle\overline{e_i + w_i^*}, \overline{e_j + w_j}\rangle^2\right) \\
\overset{②}{\le} &0.6\gamma^2\sum_{i\neq j}\left(\langle\overline{e_i + w_i}, \overline{e_j + w_j}\rangle^2 + \langle\overline{e_i + w_i^*}, \overline{e_j + w_j}\rangle^2\right) \\
\overset{③}{\le} &0.6\gamma^2\left(\frac{4\gamma}{(1-\gamma)^2} + \frac{4\gamma(1+\gamma)}{1-2\gamma}\right) < 5\gamma^3
\end{aligned}$$

where ① uses Lemma B.6.2, ② uses $\gamma \le \frac{1}{100}$, ③ uses Lemma B.6.9. So we get $\|\mathbf{\Delta}_2\|_2 \le \frac{1+\gamma}{1-\gamma}5\gamma^3 \le 6\gamma^3$. $\square$

Proof for Lemma B.2.3. Combining the results from Lemma B.7.3 and Lemma B.7.4, the lemma follows. $\square$

B.8 Proofs for Section B.3

B.8.1 Proof for Lemma B.3.1

In Lemma B.2.3, we use $\mathbf{P}(\mathbf{W})$ to approximate $-\nabla L(\mathbf{W})$ in terms of spectral norm, with approximation loss $3.5\gamma^2$. Below we will get $\mathbf{Q}(\mathbf{W})$ from $\mathbf{P}(\mathbf{W})$ by removing a few more lower order terms.

By definition 5.2.3, we have

$$\begin{aligned}
\mathbf{P}_{2,j} =& g\overline{e_j + w_j} - (\|e_j + w_j^*\|_2 - \|e_j + w_j\|_2)\overline{e_j + w_j} + \left(\mathbf{I} - \frac{1}{2}\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right)\mathbf{A}\overline{e_j + w_j} \\
&+ \left(\mathbf{I} - \frac{1}{2}\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right)(e_j + w_j) - \left(\mathbf{I} - \frac{1}{2}\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right)(e_j + w_j^*)\overline{e_j + w_j^*}^\top\overline{e_j + w_j} \\
=& g\overline{e_j + w_j} - (\|e_j + w_j^*\|_2 - \|e_j + w_j\|_2)\overline{e_j + w_j} + \left(\mathbf{I} - \frac{1}{2}\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right)\mathbf{A}\overline{e_j + w_j} \\
&+ \frac{1}{2}(e_j + w_j) - (e_j + w_j^*)\overline{e_j + w_j^*}^\top\overline{e_j + w_j} + \frac{1}{2}\overline{e_j + w_j}\|e_j + w_j^*\|_2(\overline{e_j + w_j^*}^\top\overline{e_j + w_j})^2 \\
=& g\overline{e_j + w_j} + \left(\mathbf{I} - \frac{1}{2}\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right)\mathbf{A}\overline{e_j + w_j} + \frac{3}{2}(e_j + w_j) - \overline{e_j + w_j^*}^\top\overline{e_j + w_j}(e_j + w_j^*) \\
&+ \left(\frac{1}{2}\|e_j + w_j^*\|_2(\overline{e_j + w_j^*}^\top\overline{e_j + w_j})^2 - \|e_j + w_j^*\|_2\right)\overline{e_j + w_j} \\
=& g\overline{e_j + w_j} + \left(\mathbf{I} - \frac{1}{2}\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right)\mathbf{A}\overline{e_j + w_j} - w_j^* + w_j + (1 - \overline{e_j + w_j^*}^\top\overline{e_j + w_j})(e_j + w_j^*) \\
&+ \left(\frac{1}{2}\|e_j + w_j\|_2 + \frac{1}{2}\|e_j + w_j^*\|_2(\overline{e_j + w_j^*}^\top\overline{e_j + w_j})^2 - \|e_j + w_j^*\|_2\right)\overline{e_j + w_j}
\end{aligned}$$

Combining every column together, we get

$$\mathbf{P}_2 = g\overline{\mathbf{I} + \mathbf{W}} + \mathbf{A}\overline{\mathbf{I} + \mathbf{W}} - \frac{1}{2}\overline{\mathbf{I} + \mathbf{W}}\mathrm{Diag}(\{\overline{e_j + w_j}^\top\mathbf{A}\overline{e_j + w_j}\}_{j=1}^d) - (\mathbf{W}^* - \mathbf{W}) + \overline{\mathbf{I} + \mathbf{W}^*}\mathbf{\Sigma}_1 + \overline{\mathbf{I} + \mathbf{W}}\mathbf{\Sigma}_2$$

where

$$\mathbf{\Sigma}_1 = \mathrm{Diag}(\{(\|e_j + w_j^*\|_2 - \|e_j + w_j^*\|_2\overline{e_j + w_j^*}^\top\overline{e_j + w_j})\}_{j=1}^d)$$

$$\mathbf{\Sigma}_2 = \mathrm{Diag}(\{\frac{1}{2}\|e_j + w_j\|_2 + \frac{1}{2}\|e_j + w_j^*\|_2(\overline{e_j + w_j^*}^\top \overline{e_j + w_j})^2 - \|e_j + w_j^*\|_2\}_{j=1}^d)$$

Using Lemma B.6.12, we replace $\overline{e_j + w_j}^\top \mathbf{A}\overline{e_j + w_j}$ with $e_j^\top \mathbf{A} e_j$. By Lemma B.6.1,

$$\left\|\mathbf{P}_2 - \left[g\overline{\mathbf{I} + \mathbf{W}} + \mathbf{A}\overline{\mathbf{I} + \mathbf{W}} - \frac{1}{2}\overline{\mathbf{I} + \mathbf{W}}\mathrm{Diag}(\mathbf{A}) - (\mathbf{W}^* - \mathbf{W}) + \overline{\mathbf{I} + \mathbf{W}^*}\mathbf{\Sigma}_1 + \overline{\mathbf{I} + \mathbf{W}}\mathbf{\Sigma}_2\right]\right\|_2 \leq \frac{5(1+\gamma)}{2(1-\gamma)} < 2.6\gamma^2$$

We then focus on the middle two summands in the sum.

$$\mathbf{A}\overline{\mathbf{I} + \mathbf{W}} - \frac{1}{2}\overline{\mathbf{I} + \mathbf{W}}\mathrm{Diag}(\mathbf{A}) = (\mathbf{A} - \frac{1}{2}\mathrm{Diag}(\mathbf{A})) + \mathbf{A}(\overline{\mathbf{I} + \mathbf{W}} - \mathbf{I}) - \frac{1}{2}(\overline{\mathbf{I} + \mathbf{W}} - \mathbf{I})\mathrm{Diag}(\mathbf{A})$$

By Lemma B.6.10, $\|\mathrm{Diag}(\mathbf{A})\|_2 \leq \|\mathbf{A}\|_2$, so

$$\left\|\left[\mathbf{A}\overline{\mathbf{I} + \mathbf{W}} - \frac{1}{2}\overline{\mathbf{I} + \mathbf{W}}\mathrm{Diag}(\mathbf{A})\right] - \left[\mathbf{A} - \frac{1}{2}\mathrm{Diag}(\mathbf{A})\right]\right\|_2 = \left\|\mathbf{A}(\overline{\mathbf{I} + \mathbf{W}} - \mathbf{I}) - \frac{1}{2}(\overline{\mathbf{I} + \mathbf{W}} - \mathbf{I})\mathrm{Diag}(\mathbf{A})\right\|_2$$
$$\leq \|\mathbf{A}\|_2\|\overline{\mathbf{I} + \mathbf{W}} - \mathbf{I}\|_2 + \frac{1}{2}\|\overline{\mathbf{I} + \mathbf{W}} - \mathbf{I}\|_2\|\mathrm{Diag}(\mathbf{A})\|_2 \overset{①}{\leq} \frac{3\gamma(\gamma^2+3)}{1-\gamma^2}2.05\gamma < 18.5\gamma^2$$

where ① uses Lemma B.6.11 and Lemma B.6.15.

Moreover, by Lemma B.5.1 term 2, we know $\|\mathbf{\Sigma}_1\|_2 \leq \max_{i\in[d]}(1+\gamma)\frac{\|w_i^* - w_i\|_2^2}{2(1-2\gamma)} \leq 2.07\gamma^2$, and in $\mathbf{\Sigma}_2$,

$$\left|\frac{1}{2}\|e_j + w_j^*\|_2(\overline{e_j + w_j^*}^\top \overline{e_j + w_j})^2 - \frac{1}{2}\|e_j + w_j^*\|_2\right| \leq \frac{1}{2}(1+\gamma)\left|\overline{e_j + w_j^*}^\top \overline{e_j + w_j} - 1\right|\left|\overline{e_j + w_j^*}^\top \overline{e_j + w_j} + 1\right| \leq 2.07\gamma^2$$

so the following terms approximates $\mathbf{P}_2$ with approximation loss $(2.6+18.5+2.07+2.07)\gamma^2 < 25.3\gamma^2$.

$$\overline{\mathbf{I} + \mathbf{W}}(g\mathbf{I} - \mathbf{\Sigma}_3) + \mathbf{A} - \frac{1}{2}\mathrm{Diag}(\mathbf{A}) - (\mathbf{W}^* - \mathbf{W})$$

where $\mathbf{\Sigma}_3 = \mathrm{Diag}(\{\frac{1}{2}\|e_j + w_j^*\|_2 - \frac{1}{2}\|e_j + w_j\|_2\}_{j=1}^d)$.

By Lemma B.6.16 and Corollary B.6.17, we know $\|\mathbf{A} - [\mathbf{W}^* - \mathbf{W} + (\mathbf{W}^* - \mathbf{W})^\top - \mathrm{Diag}(\mathbf{W}^* - \mathbf{W})]\|_2 \leq 9.2\gamma^2$ and $\|\mathrm{Diag}(\mathbf{A}) - \mathrm{Diag}(\mathbf{W}^* - \mathbf{W})\|_2 \leq 9.2\gamma^2$. Therefore, with approximation loss of $18.4\gamma^2$, we get

$$\left\|\left[\mathbf{A} - \frac{1}{2}\mathrm{Diag}(\mathbf{A})\right] - \left[\mathbf{W}^* - \mathbf{W} + (\mathbf{W}^* - \mathbf{W})^\top - \frac{3}{2}\mathrm{Diag}(\mathbf{W}^* - \mathbf{W})\right]\right\|_2 \leq 18.4\gamma^2$$

We then approximate $\mathbf{\Sigma}_3$:

$$\|(\overline{\mathbf{I}+\mathbf{W}})\mathbf{\Sigma}_3 - (\overline{\mathbf{I}+\mathbf{W}})\frac{1}{2}\text{Diag}(\mathbf{W}^*-\mathbf{W})\|_2 \le \frac{1+\gamma}{1-\gamma}\left(\frac{1}{2}\max_j \|\|e_j+w_j^*\|_2 - \|e_j+w_j\|_2 - w_{j,j}^* + w_{j,j}|\right) < 3.1\gamma^2$$

where the last inequality is by Lemma B.6.13. Moreover,

$$\begin{aligned}&\|\overline{\mathbf{I}+\mathbf{W}}\left(\frac{1}{2}\text{Diag}(\mathbf{W}^*-\mathbf{W})\right) - \frac{1}{2}\text{Diag}(\mathbf{W}^*-\mathbf{W})\|_2\\ \le&\|\overline{\mathbf{I}+\mathbf{W}} - \mathbf{I}\|_2\left\|\frac{1}{2}\text{Diag}(\mathbf{W}^*-\mathbf{W})\right\|_2 < 2.05\gamma\left(\frac{1}{2}\max_i |w_{i,i}^* - w_{i,i}|\right) < 2.05\gamma^2\end{aligned}$$

Putting everything together, with approximation loss of $(25.3 + 18.4 + 3.1 + 2.05)\gamma^2 = 49\gamma^2$ to $\mathbf{P}_2$, we get

$$(\mathbf{W}^*-\mathbf{W})^\top - 2\text{Diag}(\mathbf{W}^*-\mathbf{W}) + g\overline{\mathbf{I}+\mathbf{W}}$$

For $\mathbf{P}_3$, using the same idea in the proof of Lemma B.4.3, we have

$$\mathbf{P}_3 = \frac{\pi}{2}(\mathbf{W}^*-\mathbf{W}) + \left(\overline{\mathbf{I}+\mathbf{W}} - \overline{\mathbf{I}+\mathbf{W}^*}\right)\mathbf{\Sigma}_4 + \overline{\mathbf{I}+\mathbf{W}}\mathbf{\Sigma}_5$$

where $\mathbf{\Sigma}_4 = \text{Diag}(\{\theta_{j,j^*}\|e_j+w_j^*\|_2\}_{j=1}^d)$, $\mathbf{\Sigma}_5 = \text{Diag}(\{\|e_j+w_j^*\|_2 \sin\theta_{j,j^*} - \theta_{j,j^*}\|e_j+w_j^*\|_2\}_{j=1}^d)$. By Taylor's Theorem, we know $\|\mathbf{\Sigma}_5\|_2 \le \|\text{Diag}(\{\|e_j+w_j^*\|_2\theta_{j,j^*}^3/3\}_{j=1}^d)\|_2$.

Notice that $\theta_{j,j^*} \le 2.002\gamma$ by Lemma B.5.1 term 3, and $\|\overline{\mathbf{I}+\mathbf{W}} - \overline{\mathbf{I}+\mathbf{W}^*}\|_2 \le \frac{1+\gamma}{1-\gamma} - \frac{1-\gamma}{1+\gamma} \le 4.001\gamma$.

Consequently,

$$\begin{aligned}&\left\|\mathbf{P}_3 - \frac{\pi}{2}(\mathbf{W}^*-\mathbf{W})\right\|_2 \le \|\left(\overline{\mathbf{I}+\mathbf{W}} - \overline{\mathbf{I}+\mathbf{W}^*}\right)\mathbf{\Sigma}_4\|_2 + \|\overline{\mathbf{I}+\mathbf{W}}\mathbf{\Sigma}_5\|_2\\ <&4.001 * 2.002(1+\gamma)\gamma^2 + \frac{(1+\gamma)^2}{3(1-\gamma)}(2.002\gamma)^3 < 8.1\gamma^2 + 2.8\gamma^3 < 8.2\gamma^2\end{aligned}$$

we only need to keep the term $\frac{\pi}{2}(\mathbf{W}^*-\mathbf{W})$ with approximation loss $8.2\gamma^2$ to $\mathbf{P}_3$.

Now, combining the approximations to $\mathbf{P}_2$ and $\mathbf{P}_3$, and Lemma B.2.3, we have the following matrix with $(49 + 8.2 + 3.5)\gamma^2 < 61\gamma^2$ approximation loss to $-\nabla\mathsf{L}(\mathbf{W})$:

$$\frac{\pi}{2}(\mathbf{W}^*-\mathbf{W})(\mathbf{I}+uu^\top) + (\mathbf{W}^*-\mathbf{W})^\top - 2\text{Diag}(\mathbf{W}^*-\mathbf{W}) + g\overline{\mathbf{I}+\mathbf{W}}$$

where u is the all 1 vector.

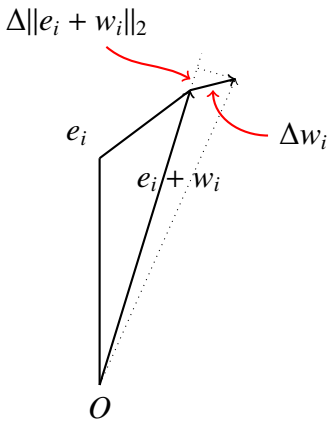


Figure B.3: Δg is approximately (the summation of) the projection of Δw_i onto $\overline{e_i + w_i}$

B.8.2 Proof for Lemma B.3.2

By Lemma B.6.4, we know $|g| \le 2d\gamma$. Using Lemma B.3.1,

$$\|\nabla \mathsf{L}(\mathbf{W})\|_2 \le 61\gamma^2 + \left\|\frac{\pi}{2}(\mathbf{W}^* - \mathbf{W})\left(\mathbf{I} + uu^\top\right) + (\mathbf{W}^* - \mathbf{W})^\top - 2\mathrm{Diag}(\mathbf{W}^* - \mathbf{W}) + g\overline{\mathbf{I} + \mathbf{W}}\right\|_2$$
$$\le 61\gamma^2 + (d+1)\pi\gamma + 2\gamma + 4\gamma + |g|\frac{1+\gamma}{1-\gamma} < 61\gamma^2 + (d+3)\pi\gamma + 2.05d\gamma < 6d\gamma.$$

B.8.3 Proof for Lemma B.3.3

In this proof, we use w_j to represent the j-th column of $\mathbf{W}_t$, and denote $\triangle w_j$ as the j-th column of $\mathbf{G}_t$.

$\Delta g_t \approx \eta\langle \mathsf{L}(\mathbf{W}_t), \overline{\mathbf{I} + \mathbf{W}_t}\rangle$

For the intuition of this section, see Figure B.3. The changes in potential function g is essentially the changes in $\|e_i + w_i\|_2$ (summing over i), which is approximately Δw_i projected onto $\overline{e_i + w_i}$. If we write it in matrix form, we get $\Delta g_t \approx \eta\langle \mathsf{L}(\mathbf{W}_t), \overline{\mathbf{I} + \mathbf{W}_t}\rangle$.

By definition we know $\|\mathbf{G}_t\|_2 = \|\nabla \mathsf{L}(\mathbf{W}_t) + \mathbf{E}_t\|_2 \overset{①}{\leq} \|\nabla \mathsf{L}(\mathbf{W}_t)\|_2 + \|\mathbf{E}_t\|_2 \overset{②}{\leq} 6d\gamma + \varepsilon = G_2$, where ① uses triangle inequality, ② uses Lemma B.3.2. We have

$$\eta\|\triangle w_j\|_2 \leq \eta\|\mathbf{G}_t\|_2 \leq \frac{\gamma^2}{G_2} \leq \frac{\gamma}{6d}, \qquad \eta^2\|\triangle w_j\|_2 \leq \eta\|\mathbf{G}_t\|_2^2 \leq \gamma^2 \tag{B.11}$$

By Definition 5.2.2, we know

$$\begin{aligned}
\triangle g_t &\triangleq g_{t+1} - g_t = \sum_{j=1}^{d}\left(\frac{\langle e_j + w_j, e_j + w_j\rangle}{\|e_j + w_j\|_2} - \frac{\langle e_j + w_j - \eta\triangle w_j, e_j + w_j - \eta\triangle w_j\rangle}{\|e_j + w_j - \eta\triangle w_j\|_2}\right) \\
&= \sum_{j=1}^{d}\left(\frac{\langle e_j + w_j, e_j + w_j\rangle\|e_j + w_j - \eta\triangle w_j\|_2 - \langle e_j + w_j - \eta\triangle w_j, e_j + w_j - \eta\triangle w_j\rangle\|e_j + w_j\|_2}{\|e_j + w_j\|_2\|e_j + w_j - \eta\triangle w_j\|_2}\right) \\
&= \sum_{j=1}^{d}\left(\frac{\|e_j + w_j\|_2(\|e_j + w_j - \eta\triangle w_j\|_2 - \|e_j + w_j\|_2) + 2\eta\langle\triangle w_j, e_j + w_j\rangle - \eta^2\|\triangle w_j\|_2^2}{\|e_j + w_j - \eta\triangle w_j\|_2}\right)
\end{aligned}$$

If we project $\eta\triangle w_j$ onto the $\overline{e_j + w_j}$ direction, we get

$$\begin{aligned}
\|e_j + w_j - \eta\triangle w_j\|_2 &= \sqrt{(\|e_j + w_j\|_2 - \langle\overline{e_j + w_j}, \eta\triangle w_j\rangle)^2 + (\|\eta\triangle_j\|_2^2 - \langle\overline{e_j + w_j}, \eta\triangle w_j\rangle^2)^2} \\
\leq \sqrt{(\|e_j + w_j\|_2 - \langle\overline{e_j + w_j}, \eta\triangle w_j\rangle)^2 + \|\eta\triangle w_j\|_2^2} &\overset{①}{\leq} \|e_j + w_j\|_2 - \langle\overline{e_j + w_j}, \eta\triangle w_j\rangle + \|\eta\triangle w_j\|_2^2
\end{aligned}$$

Using (B.11), we have $\|e_j + w_j\|_2 - \langle\overline{e_j + w_j}, \eta\triangle w_j\rangle \geq \frac{1}{2}$. By taking square on both sides, we know ① holds. It is trivial to show that $\|e_j + w_j - \eta\triangle w_j\|_2 \geq \|e_j + w_j\|_2 - \langle\overline{e_j + w_j}, \eta\triangle w_j\rangle$, so we know

$$-\langle\overline{e_j + w_j}, \eta\triangle w_j\rangle \leq \|e_j + w_j - \eta\triangle w_j\|_2 - \|e_j + w_j\|_2 \leq -\langle\overline{e_j + w_j}, \eta\triangle w_j\rangle + \|\eta\triangle w_j\|_2^2 \tag{B.12}$$

Thus, with approximation loss $\sum_{j=1}^{d}\frac{\|e_j+w_j\|_2\|\eta\triangle w_j\|_2^2}{\|e_j+w_j-\eta\triangle w_j\|_2}$, we have :

$$\begin{aligned}
\triangle g_t &\approx \sum_{j=1}^{d}\left(\frac{-\|e_j + w_j\|_2\langle\overline{e_j + w_j}, \eta\triangle w_j\rangle + 2\eta\langle\triangle w_j, e_j + w_j\rangle - \eta^2\|\triangle w_j\|_2^2}{\|e_j + w_j - \eta\triangle w_j\|_2}\right) \\
&= \sum_{j=1}^{d}\frac{\eta\langle\triangle w_j, e_j + w_j\rangle - \eta^2\|\triangle w_j\|_2^2}{\|e_j + w_j - \eta\triangle w_j\|_2} \\
&= \sum_{j=1}^{d}\frac{-\eta^2\|\triangle w_j\|_2^2}{\|e_j + w_j - \eta\triangle w_j\|_2} + \sum_{j=1}^{d}\frac{(\|e_j + w_j\|_2 - \|e_j + w_j - \eta\triangle w_j\|_2)\eta\langle\triangle w_j, \overline{e_j + w_j}\rangle}{\|e_j + w_j - \eta\triangle w_j\|_2} + \eta\langle\mathbf{G}_t, \overline{\mathbf{I} + \mathbf{W}_t}\rangle
\end{aligned}$$

Thus we get the following approximation for $\triangle g_t$.

$$
\begin{aligned}
&|\triangle g_t - \eta\langle \mathbf{G}_t, \overline{\mathbf{I}+\mathbf{W}_t}\rangle| \\
\leq &\sum_{j=1}^{d}\left|\frac{-\eta^2\|\triangle w_j\|_2^2}{\|e_j+w_j-\eta\triangle w_j\|_2} + \frac{(\|e_j+w_j\|_2 - \|e_j+w_j-\eta\triangle w_j\|_2)\eta\langle \triangle w_j, \overline{e_j+w_j}\rangle}{\|e_j+w_j-\eta\triangle w_j\|_2} + \frac{\|e_j+w_j\|_2\|\eta\triangle w_j\|_2^2}{\|e_j+w_j-\eta\triangle w_j\|_2}\right| \\
\overset{①}{\leq} &\sum_{j=1}^{d}\left[\left|\frac{\eta\langle \triangle w_j, \overline{e_j+w_j}\rangle(\eta\langle \triangle w_j, \overline{e_j+w_j}\rangle + \|\eta\triangle w_j\|_2^2)}{\|e_j+w_j-\eta\triangle w_j\|_2}\right| + 0.02\eta^2\|\triangle w_j\|_2^2\right] \\
\overset{②}{\leq} &\sum_{j=1}^{d}\left[\frac{\eta^2\|\triangle w_j\|_2^2 + \eta^3\|\triangle w_j\|_2^3}{\|e_j+w_j-\eta\triangle w_j\|_2} + 0.02\eta\gamma^2\right] \overset{③}{\leq} 1.04\eta d\gamma^2
\end{aligned}
$$

where ① uses (B.12) again, and ② ③ uses (B.11), $\gamma \leq \frac{1}{100}$ and $\|e_j + w_j - \eta\triangle w_j\|_2 \geq 0.98$.

Thus $|\triangle g_t - \eta\langle \nabla \mathsf{L}(\mathbf{W}_t), \overline{\mathbf{I}+\mathbf{W}_t}\rangle| \leq 1.04\eta d\gamma^2 + |\eta\langle \mathbf{E}_t, \overline{\mathbf{I}+\mathbf{W}_t}\rangle| < 1.04\eta d\gamma^2 + 1.03\eta\sqrt{d}\varepsilon$

$\Delta g_t \approx \eta \operatorname{Tr}(\nabla \mathsf{L}(\mathbf{W}_t))$

We want to approximate $\overline{\mathbf{I}+\mathbf{W}_t}$ with $\mathbf{I}$. Below is the error bound.

$$
\begin{aligned}
&|\langle \nabla \mathsf{L}(\mathbf{W}_t), \overline{\mathbf{I}+\mathbf{W}_t} - \mathbf{I}\rangle| = |\langle \nabla \mathsf{L}(\mathbf{W}_t) + \mathbf{Q}_t - \mathbf{Q}_t, \overline{\mathbf{I}+\mathbf{W}_t} - \mathbf{I}\rangle| \\
\overset{①}{=} &d\cdot 61\gamma^2\cdot 2.05\gamma + \sum_{i=1}^{d} 2.05\gamma\left\|(\mathbf{Q}_t - \frac{\pi}{2}(\mathbf{W}^* - \mathbf{W}_t)uu^\top)_i\right\|_2 + \left\langle \frac{\pi}{2}(\mathbf{W}^* - \mathbf{W}_t)uu^\top, \overline{\mathbf{I}+\mathbf{W}_t} - \mathbf{I}\right\rangle \\
\overset{②}{\leq} &1.251d\gamma^2 + 2.05d\gamma\left(\pi\gamma + 2\gamma + 4\gamma + \frac{1+\gamma}{1-\gamma}|g_t|\right) + \operatorname{Tr}\left(\left[\frac{\pi}{2}(\mathbf{W}^* - \mathbf{W}_t)u\right]\left[u^\top\overline{\mathbf{I}+\mathbf{W}_t} - \mathbf{I}\right]^\top\right) \\
\overset{③}{\leq} &20d\gamma^2 + 2.1d\gamma|g_t| + \left\|\frac{\pi}{2}(\mathbf{W}^* - \mathbf{W}_t)u\right\|_2\left\|(\overline{\mathbf{I}+\mathbf{W}_t} - \mathbf{I})u\right\|_2 \overset{④}{\leq} 20d\gamma^2 + 2.1d\gamma|g_t| + \frac{2.05\pi}{2}\|s\|_2\gamma\sqrt{d}
\end{aligned}
$$

where ① uses Cauchy Schwartz and Lemma B.6.15, ② uses the definition of $\mathbf{Q}$ and Lemma B.6.1, ③ holds as for any vector u, v, $\operatorname{Tr}(uv^\top) \leq \|u\|_2\|v\|_2$, ④ uses Lemma B.6.15.

Hence,

$$|\triangle g_t - \eta\langle \nabla \mathsf{L}(\mathbf{W}_t), \mathbf{I}\rangle|$$

$$
\begin{aligned}
&\leq 1.04\eta d\gamma^2 + 1.03\eta\sqrt{d}\varepsilon + |\eta\langle\nabla \mathsf{L}(\mathbf{W}_t), \overline{\mathbf{I}+\mathbf{W}_t} - \mathbf{I}\rangle| \\
&< 1.04\eta d\gamma^2 + 1.03\eta\sqrt{d}\varepsilon + 20\eta d\gamma^2 + 2.1\eta d\gamma|g_t| + \frac{2.05\pi}{2}\eta\|s\|_2\gamma\sqrt{d} \\
&< 21.1\eta d\gamma^2 + 1.03\eta\sqrt{d}\varepsilon + 2.1\eta d\gamma|g_t| + \frac{2.05\pi}{2}\eta\|s\|_2\gamma\sqrt{d}
\end{aligned}
$$

So with approximation loss of $21.1\eta d\gamma^2 + 1.03\eta\sqrt{d}\varepsilon + 2.1\eta d\gamma|g_t| + \frac{2.05\pi}{2}\eta\|s\|_2\gamma\sqrt{d}$, it suffices to consider $\eta\operatorname{Tr}(\nabla \mathsf{L}(\mathbf{W}_t))$.

$\Delta g_t \approx -\eta(d + \frac{\pi}{2} - 1)g_t$

According to Lemma B.3.1, with approximation loss of $61\gamma^2$, we can use $-\mathbf{Q}_t$ to approximate $\nabla \mathsf{L}(\mathbf{W}_t)$.

$$
\begin{aligned}
&\operatorname{Tr}(\mathbf{Q}_t) = \frac{\pi}{2}\operatorname{Tr}\left((\mathbf{W}^* - \mathbf{W}_t)\left(\mathbf{I} + uu^\top\right)\right) + \operatorname{Tr}(\mathbf{W}^* - \mathbf{W}_t)^\top - 2\operatorname{Tr}(\operatorname{Diag}(\mathbf{W}^* - \mathbf{W}_t)) + g\operatorname{Tr}(\overline{\mathbf{I}+\mathbf{W}_t}) \\
=&\left(\frac{\pi}{2} - 1\right)\operatorname{Tr}(\mathbf{W}^* - \mathbf{W}_t) + \frac{\pi}{2}\operatorname{Tr}\left((\mathbf{W}^* - \mathbf{W}_t)\left(uu^\top\right)\right) + g\operatorname{Tr}(\overline{\mathbf{I}+\mathbf{W}_t}) \\
=&\left(\frac{\pi}{2} - 1\right)(\operatorname{Tr}(\mathbf{W}^* - \mathbf{W}_t) - g_t) + \left(\frac{\pi}{2} - 1\right)g_t + \frac{\pi}{2}\operatorname{Tr}\left((\mathbf{W}^* - \mathbf{W}_t)\left(uu^\top\right)\right) + g_t\operatorname{Tr}(\overline{\mathbf{I}+\mathbf{W}_t})
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
&\left|\operatorname{Tr}(\mathbf{Q}_t) - g_t\operatorname{Tr}(\mathbf{I}) - \left(\frac{\pi}{2} - 1\right)g_t\right| = \left|\operatorname{Tr}(\mathbf{Q}_t) - \left(d + \frac{\pi}{2} - 1\right)g_t\right| \\
\leq&\left|\left(\frac{\pi}{2} - 1\right)(\operatorname{Tr}(\mathbf{W}^* - \mathbf{W}_t) - g_t) + \frac{\pi}{2}\operatorname{Tr}\left((\mathbf{W}^* - \mathbf{W}_t)\left(uu^\top\right)\right) + g_t(\operatorname{Tr}(\overline{\mathbf{I}+\mathbf{W}_t} - \mathbf{I}))\right| \\
\overset{①}{\leq}&6.07\left(\frac{\pi}{2} - 1\right)d\gamma^2 + \frac{\pi}{2}\|s_t\|_2\sqrt{d} + 2.05|g_t|d\gamma
\end{aligned}
$$

where ① uses Lemma B.6.14 and Lemma B.6.15. Thus,

$$
\begin{aligned}
&\left|\triangle g_t - \left[-\eta\left(d + \frac{\pi}{2} - 1\right)g_t\right]\right| \\
\leq&\eta\left[21.1d\gamma^2 + 1.03\sqrt{d}\varepsilon + 2.1d\gamma|g_t| + \frac{2.05\pi}{2}\|s\|_2\gamma\sqrt{d} + 61d\gamma^2 + 2.05|g_t|d\gamma + 6.07\left(\frac{\pi}{2} - 1\right)d\gamma^2 + \frac{\pi}{2}\|s_t\|_2\sqrt{d}\right]
\end{aligned}
$$

$$\leq \eta\left[86d\gamma^2 + 1.03\sqrt{d}\varepsilon + 4.15d\gamma|g_t| + 4.8\|s_t\|_2\gamma\sqrt{d}\right]$$

Now we have

$$|g_{t+1}| = |g_t + \triangle g_t| \leq \left(1 - \eta\left(d + \frac{\pi}{2} - 1 - 4.15d\gamma\right)\right)|g_t| + 86\eta d\gamma^2 + 1.03\eta\sqrt{d}\varepsilon + 4.8\eta\|s_t\|_2\gamma\sqrt{d}$$
$$\leq (1 - 0.95\eta d)|g_t| + 86\eta d\gamma^2 + 1.03\eta\sqrt{d}\varepsilon + 4.8\eta\|s_t\|_2\gamma\sqrt{d}$$

B.8.4 Proof for Lemma B.3.4

By definition of s_t,

$$\triangle s_t \triangleq s_{t+1} - s_t = (\mathbf{W}_t - \mathbf{W}_{t+1})u = \eta(\nabla \mathsf{L}(\mathbf{W}_t) + \mathbf{E}_t)u = -\eta\mathbf{Q}_t u + \eta(\mathbf{Q}_t + \nabla \mathsf{L}(\mathbf{W}_t) + \mathbf{E}_t)u$$

By definition of $\mathbf{Q}_t$,

$$\begin{aligned}\mathbf{Q}_t u =& \left(\frac{\pi}{2}(\mathbf{W}^* - \mathbf{W}_t)\left(\mathbf{I} + uu^\top\right) + (\mathbf{W}^* - \mathbf{W}_t)^\top - 2\mathrm{Diag}(\mathbf{W}^* - \mathbf{W}_t) + g_t\overline{\mathbf{I} + \mathbf{W}_t}\right)u \\ =& \frac{(d+1)\pi}{2}s_t + \left((\mathbf{W}^* - \mathbf{W}_t)^\top - 2\mathrm{Diag}(\mathbf{W}^* - \mathbf{W}_t) + g_t\overline{\mathbf{I} + \mathbf{W}_t}\right)u\end{aligned}$$

Thus, we know

$$\begin{aligned}&\left\|\mathbf{Q}_t u - \frac{(d+1)\pi}{2}s_t\right\|_2 = \left\|\left((\mathbf{W}^* - \mathbf{W}_t)^\top - 2\mathrm{Diag}(\mathbf{W}^* - \mathbf{W}_t) + g_t\overline{\mathbf{I} + \mathbf{W}_t}\right)u\right\|_2 \\ \leq& \sqrt{d}\left(\|(\mathbf{W}^* - \mathbf{W}_t)^\top\|_2 + 2\|\mathrm{Diag}(\mathbf{W}^* - \mathbf{W}_t)\|_2 + \|g_t\overline{\mathbf{I} + \mathbf{W}_t}\|_2\right) \\ \overset{①}{\leq}& \sqrt{d}\left(2\gamma + 4\gamma + |g_t|\frac{1+\gamma}{1-\gamma}\right) < (6\gamma + 1.03|g_t|)\sqrt{d}\end{aligned}$$

where ① uses Lemma B.6.1 and Lemma B.6.10.

By Lemma B.3.1, $\|\triangle s_t - [-\eta\frac{(d+1)\pi}{2}s_t]\|_2 < \eta(6\gamma + 1.03|g_t|)\sqrt{d} + \eta\|(\mathbf{Q}_t + \nabla \mathsf{L}(\mathbf{W}_t) + \mathbf{E}_t)u\|_2 \leq \eta(6.61\gamma + 1.03|g_t| + \varepsilon)\sqrt{d}$.

B.8.5 Proof for Lemma B.3.5

Combining Lemma B.3.3 and Lemma B.3.4, we get

$$
\begin{aligned}
&|g_{t+1}| + \|s_{t+1}\|_2 \\
\leq&(1 - 0.95\eta d)(|g_t| + \|s_t\|_2) + \eta(6.6\gamma + 1.03|g_t| + \varepsilon)\sqrt{d} + 86\eta d\gamma^2 + 1.03\eta\sqrt{d}\varepsilon + (4.8\eta\gamma\sqrt{d} - 0.62\eta d)\|s_t\|_2 \\
\overset{①}{\leq}&(1 - 0.95\eta d)(|g_t| + \|s_t\|_2) + 6.6\eta\gamma\sqrt{d} + 86\eta d\gamma^2 + \eta 1.03|g_t|\sqrt{d} + 2.03\eta\sqrt{d}\varepsilon \\
\overset{②}{\leq}&(1 - 0.84\eta d)(|g_t| + \|s_t\|_2) + 6.6\eta\gamma\sqrt{d} + 87\eta d\gamma^2
\end{aligned}
$$

where ① uses $\gamma \leq \frac{1}{100}$, $d \geq 100$, ② uses $\varepsilon \leq \gamma^2$ and $d \geq 100$. So if the following inequality holds, $|g_t| + \|s_t\|_2$ will always decrease by factor at least $1 - 0.5\eta d$.

$$
0.34\eta d(|g_t| + \|s_t\|_2) \geq 6.6\eta\gamma\sqrt{d} + 87\eta d\gamma^2
$$

Which gives

$$
|g_t| + \|s_t\|_2 \geq \frac{6.6\eta\gamma\sqrt{d} + 87\eta d\gamma^2}{0.34\eta d} = \frac{6.6\gamma}{0.34\sqrt{d}} + \frac{87\gamma^2}{0.34}
$$

where the last expression is smaller than 4.5γ. Hence, $|g_t| + \|s_t\|_2$ will keep decreasing by $1 - 0.5\eta d$ as long as it is larger than 4.5γ. So we have $\|s_t\|_2 \leq 4.5\gamma$. Now plug it back to the updating rule of $|g_t|$:

$$
\begin{aligned}
|g_{t+1}| \leq&(1 - 0.95\eta d)|g_t| + 86\eta d\gamma^2 + 1.03\eta\sqrt{d}\varepsilon + 4.8\eta\|s_t\|_2\gamma\sqrt{d} \\
\leq&(1 - 0.95\eta d)|g_t| + 86\eta d\gamma^2 + 1.03\eta\sqrt{d}\varepsilon + 21.6\eta\gamma^2\sqrt{d}
\end{aligned}
$$

In order to get factor $1 - 0.5\eta d$, we have

$$
0.45\eta d|g_t| \geq 86\eta d\gamma^2 + 1.03\eta\sqrt{d}\varepsilon + 21.6\eta\gamma^2\sqrt{d}
$$

Solve this inequality, we get

$$
\frac{86\eta d\gamma^2 + 1.03\eta\sqrt{d}\varepsilon + 21.6\eta\gamma^2\sqrt{d}}{0.45\eta d} = \frac{86\gamma^2}{0.45} + \frac{1.03\varepsilon + 21.6\gamma^2}{0.45\sqrt{d}} \leq 197\gamma^2
$$

The last inequality uses $d \geq 100, \varepsilon \leq \gamma^2$. So even after $|g_t| + \|s_t\|_2$ is below 4.5γ, $|g_t|$ will keep decreasing by factor $1 - 0.5\eta d$ until it is smaller than $197\gamma^2$.

Finally we bound the number of steps to arrive $197\gamma^2$. Let $\gamma = \frac{1}{400}, \gamma_0 = \frac{1}{8000}$. Again, the constants here are pretty loose. Since $|g_t| \leq (1 - 0.5\eta d)^t |g_0| \leq (1 - 0.5\eta d)^t 2d\gamma_0$, in order to let $g_t \leq 197\gamma^2$, it suffices to have $t \geq \frac{\log \frac{197\gamma^2}{2d\gamma_0}}{\log(1-\frac{\eta d}{2})}$. Since ηd is small, by Taylor expansion we know $\log(1 - \frac{\eta d}{2}) \approx -\frac{\eta d}{2}$. Thus, it suffices to let $t \geq \frac{2\log(0.203d)}{\eta d}$. Notice that $\frac{\log(0.203d)}{d}$ is decreasing for $d \geq 100$, we know it suffices to let $t \geq \frac{1}{16\eta}$.

B.8.6 Proof for Lemma B.3.6

Let $\mathbf{H} = \mathbf{W} - \mathbf{W}^*$, by the updating rule of $\mathbf{W}_t$ and the definition of $\mathbf{Q}_t$, we know

$$\mathbf{H}_{t+1} = \mathbf{H}_t - \eta \mathbf{H}_t \left(\frac{\pi}{2} uu^\top + \frac{\pi}{2}\right) - \eta \mathbf{H}_t^\top + 2\eta \mathrm{Diag}(\mathbf{H}_t) + \eta g_t \overline{\mathbf{I} + \mathbf{W}} - \eta(\mathbf{G}_t + \mathbf{Q}_t)$$

That gives,

$$\begin{aligned}
&\|\mathbf{H}_{t+1} + \mathbf{H}_{t+1}^\top\|_2 \\
\leq& \left\| (\mathbf{H}_t + \mathbf{H}_t^\top)\left(\mathbf{I} - \eta\left(\frac{\pi}{2} uu^\top + \frac{\pi}{2} + 1\right)\right)\right\|_2 + 2\eta \left\|\mathrm{Diag}(\mathbf{H}_t + \mathbf{H}_t^\top)\right\|_2 + 2\eta|g_t|\|\overline{\mathbf{I} + \mathbf{W}}\|_2 + 2\eta \|\mathbf{E}_t + \nabla \mathsf{L}(\mathbf{W}_t) + \mathbf{Q}_t\|_2 \\
\overset{①}{\leq}& \left(\mathbf{I} - \eta\left(\frac{\pi}{2} + 1\right)\right)\|\mathbf{H}_t + \mathbf{H}_t^\top\|_2 + 2\eta\|\mathbf{H}_t + \mathbf{H}_t^\top\|_2 + \frac{2(1+\gamma)\eta|g_t|}{1-\gamma} + 2\eta\varepsilon + 122\eta\gamma^2 \\
\overset{②}{\leq}& \left(\mathbf{I} - \eta\left(\frac{\pi}{2} - 1\right)\right)\|\mathbf{H}_t + \mathbf{H}_t^\top\|_2 + 2.05\eta|g_t| + 124\eta\gamma^2
\end{aligned} \tag{B.13}$$

where ① uses Lemma B.6.18, Lemma B.6.10, $\|\mathbf{E}_t\|_2 \leq \varepsilon$ and Lemma B.3.1. ② uses $\varepsilon \leq \gamma^2$ and $\gamma \leq \frac{1}{100}$.

Similarly, we get

$$\|\mathbf{H}_{t+1} - \mathbf{H}_{t+1}^\top\|_2$$

$$\overset{①}{\leq} \left\|(\mathbf{H}_t - \mathbf{H}_t^\top)\left(\mathbf{I} - \eta\left(\frac{\pi}{2}uu^\top + \frac{\pi}{2} - 1\right)\right)\right\|_2 + \eta|g_t|\|\overline{\mathbf{I}+\mathbf{W}} - \mathbf{I} + \mathbf{I} - \overline{\mathbf{I}+\mathbf{W}}^\top\|_2 + 2\eta\,\|\mathbf{E}_t + \nabla \mathsf{L}(\mathbf{W}_t) + \mathbf{Q}_t\|_2$$
$$\overset{②}{\leq} \left(\mathbf{I} - \eta\left(\frac{\pi}{2} - 1\right)\right)\|\mathbf{H}_t - \mathbf{H}_t^\top\|_2 + 4.10\eta\gamma|g_t| + 124\eta\gamma^2 \tag{B.14}$$

where ① holds as the diagonal terms cancel out, ② uses Lemma B.6.18, Lemma B.6.15.

Adding (B.13) and (B.14), we get

$$\begin{aligned}&\|\mathbf{H}_{t+1} + \mathbf{H}_{t+1}^\top\|_2 + \|\mathbf{H}_{t+1} - \mathbf{H}_{t+1}^\top\|_2 \\ \leq &\left(\mathbf{I} - \eta\left(\frac{\pi}{2} - 1\right)\right)(\|\mathbf{H}_t + \mathbf{H}_t^\top\|_2 + \|\mathbf{H}_t - \mathbf{H}_t^\top\|_2) + 2.1\eta|g_t| + 248\eta\gamma^2\end{aligned} \tag{B.15}$$

For any $T > 0$, by applying (B.15) recursively, we have

$$\|\mathbf{H}_T + \mathbf{H}_T^\top\|_2 + \|\mathbf{H}_T - \mathbf{H}_T^\top\|_2 \leq \|\mathbf{H}_0 + \mathbf{H}_0^\top\|_2 + \|\mathbf{H}_0 - \mathbf{H}_0^\top\|_2 + 2.1\eta\sum_{t=0}^{T-1}|g_t| + 248\eta T\gamma^2$$

By Lemma B.6.4 we know $|g_0| \leq 2d\gamma_0$, so $2.1\eta\sum_{t=0}^{T-1}|g_t| \leq \frac{2.1\eta|g_0|(1-(1-0.5\eta d)^T)}{(0.5\eta d)} \leq \frac{4.2|g_0|}{d} \leq 8.4\gamma_0$.

By the proof of Lemma B.3.5, we know $T \leq \frac{1}{16\eta}$, so $248\eta T\gamma^2 \leq 15.5\gamma^2$.

By triangle inequality, we know $\|\mathbf{H}_0\|_2 \leq \|\mathbf{W}_0\|_2 + \|\mathbf{W}^*\|_2 \leq 2\gamma_0$, so $\|\mathbf{H}_0 + \mathbf{H}_0^\top\|_2 + \|\mathbf{H}_0 - \mathbf{H}_0^\top\|_2 \leq 4\|\mathbf{H}_0\|_2 \leq 8\gamma_0$.

By triangle inequality again we get

$$\|\mathbf{H}_T\|_2 \leq \|\mathbf{H}_T + \mathbf{H}_T^\top\|_2 + \|\mathbf{H}_T - \mathbf{H}_T^\top\|_2 \leq \|\mathbf{H}_0 + \mathbf{H}_0^\top\|_2 + \|\mathbf{H}_0 - \mathbf{H}_0^\top\|_2 + 19\gamma^2 + 8.4\gamma_0 \leq 16.4\gamma_0 + 15.5\gamma^2$$

Recall we set $\gamma = \frac{1}{400}, \gamma_0 = \frac{1}{8000}$ in the proof of Lemma B.3.5, we know $\|\mathbf{W}_T\|_2 \leq \|\mathbf{W}^*\|_2 + \|\mathbf{H}_T\|_2 \leq 17.4\gamma_0 + 15.5\gamma^2 \leq \frac{1}{440} \leq \gamma$.

B.8.7 Proof for Lemma B.3.7

First, by the proof of Lemma B.3.5, we know $|g_t|$ will keep small if $\|\mathbf{W}_t\|_2 \le \gamma \le \frac{1}{100}$.

Adding (B.13) and (B.14), we get

$$\begin{aligned}
&\|\mathbf{H}_{t+1} + \mathbf{H}_{t+1}^\top\|_2 + \|\mathbf{H}_{t+1} - \mathbf{H}_{t+1}^\top\|_2 \\
\le &\left(\mathbf{I} - \eta\left(\frac{\pi}{2} - 1\right)\right)(\|\mathbf{H}_{t+1} + \mathbf{H}_{t+1}^\top\|_2 + \|\mathbf{H}_{t+1} - \mathbf{H}_{t+1}^\top\|_2) + 2.1\eta|g_t| + 248\eta\gamma^2 \\
\overset{①}{\le} &\left(\mathbf{I} - \eta\left(\frac{\pi}{2} - 1\right)\right)(\|\mathbf{H}_{t+1} + \mathbf{H}_{t+1}^\top\|_2 + \|\mathbf{H}_{t+1} - \mathbf{H}_{t+1}^\top\|_2) + 661\eta\gamma^2 \qquad \text{(B.16)}
\end{aligned}$$

where ① holds as $|g_t| \le 197\gamma^2$. So either $\|\mathbf{H}_{t+1} + \mathbf{H}_{t+1}^\top\|_2 + \|\mathbf{H}_{t+1} - \mathbf{H}_{t+1}^\top\|_2$ keeps decreasing, or it increases, i.e.,

$$\eta\left(\frac{\pi}{2} - 1\right)(\|\mathbf{H}_{t+1} + \mathbf{H}_{t+1}^\top\|_2 + \|\mathbf{H}_{t+1} - \mathbf{H}_{t+1}^\top\|_2) \le 197\eta\gamma^2$$

That gives,

$$\|\mathbf{H}_{t+1} + \mathbf{H}_{t+1}^\top\|_2 + \|\mathbf{H}_{t+1} - \mathbf{H}_{t+1}^\top\|_2 \le \frac{197\gamma^2}{\frac{\pi}{2} - 1} \le 346\gamma^2$$

Therefore, combined with the proof of Lemma B.3.6, we know $\|\mathbf{H}_{t+1} + \mathbf{H}_{t+1}^\top\|_2 + \|\mathbf{H}_{t+1} - \mathbf{H}_{t+1}^\top\|_2$ will keep decreasing until it is at most $346\gamma^2$. Now,

$$\|\mathbf{W}_t\|_2 \le \|\mathbf{H}_t\|_2 + \|\mathbf{W}^*\|_2 \le \|\mathbf{H}_{t+1} + \mathbf{H}_{t+1}^\top\|_2 + \|\mathbf{H}_{t+1} - \mathbf{H}_{t+1}^\top\|_2 + \gamma_0 \overset{①}{\le} (346 + 20)\gamma^2 \le \gamma$$

where ① holds as $\gamma_0 = \frac{1}{8000}$. So $\|\mathbf{W}_t\|_2$ is always bounded by γ.

B.9 Proofs for Section B.4

For notational simplicity, denote

$$x_j \triangleq \left(\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right)(w_j^* - w_j),$$

$$\mathbf{X} \triangleq (x_1, \cdots, x_d) \tag{B.17}$$

$$y_j \triangleq \left(\mathbf{I} - \overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right)(w_j^* - w_j),$$

$$\mathbf{Y} \triangleq (y_1, \cdots, y_d) \tag{B.18}$$

$$z_j \triangleq \left(\mathbf{I} - \frac{1}{2}\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right)(w_j^* - w_j),$$

$$\mathbf{Z} \triangleq (z_1, \cdots, z_d)$$

We have the following relationship between x_j, y_j, z_j.

Lemma B.9.1.

$$\|z_j\|_2^2 = \frac{1}{4}\|x_j\|_2^2 + \|y_j\|_2^2, \quad \|x_j\|_2^2 + \|y_j\|_2^2 = \|w_j^* - w_j\|_2^2 \tag{B.19}$$

Proof for Lemma B.9.1. By definition,

$$\begin{aligned}
\|z_j\|_2^2 =& \|w_j^* - w_j\|_2^2 \left(\mathbf{I} - \frac{1}{2}\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right)^\top \left(\mathbf{I} - \frac{1}{2}\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right) \\
=& \|w_j^* - w_j\|_2^2 \left(\mathbf{I} - \overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top + \frac{1}{4}\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top \overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right) \\
=& \|w_j^* - w_j\|_2^2 \left(\mathbf{I} - \frac{3}{4}\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right),
\end{aligned}$$

and similarly

$$\begin{aligned}
\|y_j\|_2^2 =& \|w_j^* - w_j\|_2^2 \left(\mathbf{I} - \overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right)^\top \left(\mathbf{I} - \overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right) = \|w_j^* - w_j\|_2^2 \left(\mathbf{I} - \overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right), \\
\|x_j\|_2^2 =& \|w_j^* - w_j\|_2^2 \left(\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right)^\top \left(\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right) = \|w_j^* - w_j\|_2^2 \left(\overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top\right)
\end{aligned}$$

The lemma follows. □

B.9.1 Proof for Lemma B.4.1

In this proof, we heavily use the following trick between the summation of four vector products, and the trace of four matrix products. We give one example below, and other cases are similar.

Lemma B.9.2. $\sum_{i,j} z_j^\top (e_i + w_i^*)(\overline{e_i + w_i^*} - \overline{e_i + w_i})^\top \overline{e_j + w_j} = \mathrm{Tr}\left([\mathbf{Z}^\top (\mathbf{I} + \mathbf{W}^*)] \left[(\overline{\mathbf{I} + \mathbf{W}^*} - \overline{\mathbf{I} + \mathbf{W}})^\top \overline{\mathbf{I} + \mathbf{W}} \right] \right)$.

Proof. By definition, $\mathrm{Tr}(\mathbf{AB}) = \sum_{j=1}^d (\mathbf{AB})_{j,j} = \sum_{i,j} \mathbf{A}_{j,i} \mathbf{B}_{i,j}$. Thus,

$$\mathrm{Tr}\left([\mathbf{Z}^\top (\mathbf{I} + \mathbf{W}^*)] \left[(\overline{\mathbf{I} + \mathbf{W}^*} - \overline{\mathbf{I} + \mathbf{W}})^\top \overline{\mathbf{I} + \mathbf{W}} \right] \right) = \sum_{i,j} [\mathbf{Z}^\top (\mathbf{I} + \mathbf{W}^*)]_{j,i} \left[(\overline{\mathbf{I} + \mathbf{W}^*} - \overline{\mathbf{I} + \mathbf{W}})^\top \overline{\mathbf{I} + \mathbf{W}} \right]_{i,j}$$

By definition, $[\mathbf{Z}^\top (\mathbf{I} + \mathbf{W}^*)]_{j,i} = z_j^\top (e_i + w_i^*)$, and $\left[(\overline{\mathbf{I} + \mathbf{W}^*} - \overline{\mathbf{I} + \mathbf{W}})^\top \overline{\mathbf{I} + \mathbf{W}} \right]_{i,j} = (\overline{e_i + w_i^*} - \overline{e_i + w_i})^\top \overline{e_j + w_j}$, so the lemma follows. □

Now we proceed to prove Lemma B.4.1. We first bound $\sum_{j=1}^d z_j^\top \mathbf{A}_j \overline{e_j + w_j}$ below by splitting $\mathbf{A}_j$ into three parts, and then improve the lower bound in Lemma B.9.4.

Lemma B.9.3. *If* $\|\mathbf{W}\|_2, \|\mathbf{W}^*\|_2 \le \gamma \le \frac{1}{100}$, *we have*

$$\sum_{j=1}^d z_j^\top \mathbf{A}_j \overline{e_j + w_j} \ge -8\gamma \|\mathbf{W}^* - \mathbf{W}\|_F^2 - \sqrt{\|\mathbf{W}^* - \mathbf{W}\|_f^2 - \frac{3}{4}\|\mathbf{X}\|_F^2} \sqrt{\|\mathbf{W}^* - \mathbf{W}\|_F^2 - \|\mathbf{X}\|_F^2}$$

.

Proof. We rewrite $\mathbf{A}_j$ as

$$\mathbf{A}_j = \mathbf{B}_j + \frac{1}{2}\mathbf{C}_j + \mathbf{D}_j \tag{B.20}$$

where

$$\mathbf{B}_j = \sum_{i \ne j} (e_i + w_i^*)(\overline{e_i + w_i^*} - \overline{e_i + w_i})^\top, \quad \mathbf{C}_j = \sum_{i \ne j} \langle w_i^* - w_i, \overline{e_i + w_i} \rangle \overline{e_i + w_i} \cdot \overline{e_i + w_i}^\top, \quad \mathbf{D}_j = \left(\sum_{i \ne j} z_i \overline{e_i + w_i}^\top \right)$$

For notational simplicity, we also write $\mathbf{B}, \mathbf{C}, \mathbf{D}$ as the corresponding terms with sum $\sum_{i=1}^{d}$ instead of $\sum_{i\neq j}$, so they do not depend on index j. We estimate $\mathbf{B}, \mathbf{C}, \mathbf{D}$ first, then estimate $\mathbf{B}_j, \mathbf{C}_j, \mathbf{D}_j$ respectively by taking the differences.

1. From $\mathbf{B}$ to $\mathbf{B}_j$:

$$\sum_{j=1}^{d} z_j^\top \mathbf{B}\overline{e_j + w_j} = \sum_{i,j} z_j^\top (e_i + w_i^*)(\overline{e_i + w_i^*} - \overline{e_i + w_i})^\top \overline{e_j + w_j}$$

$$\overset{①}{=} \mathrm{Tr}\left([\mathbf{Z}^\top(\mathbf{I}+\mathbf{W})]\left[(\overline{\mathbf{I}+\mathbf{W}^*} - \overline{\mathbf{I}+\mathbf{W}})^\top \overline{\mathbf{I}+\mathbf{W}}\right]\right) \overset{②}{\geq} -\left\|(\mathbf{I}+\mathbf{W})^\top \mathbf{Z}\right\|_F \left\|\overline{\mathbf{I}+\mathbf{W}}^\top (\overline{\mathbf{I}+\mathbf{W}^*} - \overline{\mathbf{I}+\mathbf{W}})\right\|_F$$

$$\overset{③}{\geq} -\|\mathbf{I}+\mathbf{W}\|_2 \|\overline{\mathbf{I}+\mathbf{W}}\|_2 \|\mathbf{Z}\|_F \left\|\overline{\mathbf{I}+\mathbf{W}^*} - \overline{\mathbf{I}+\mathbf{W}}\right\|_F \overset{④}{\geq} -\frac{(1+\gamma)^2}{1-\gamma}\|\mathbf{Z}\|_F \left\|\overline{\mathbf{I}+\mathbf{W}^*} - \overline{\mathbf{I}+\mathbf{W}}\right\|_F \tag{B.21}$$

where ① uses Lemma B.9.2, ② uses $\mathrm{Tr}(\mathbf{A}\mathcal{B}) \geq -\|\mathbf{A}\|_F\|\mathcal{B}\|_F$, ③ uses $\|\mathbf{A}\mathcal{B}\|_F \leq \|\mathbf{A}\|_2\|\mathcal{B}\|_F$, and ④ uses Lemma B.6.1. By Lemma B.5.1 term 1, we have

$$\left\|\overline{\mathbf{I}+\mathbf{W}^*} - \overline{\mathbf{I}+\mathbf{W}}\right\|_F \leq \sqrt{\frac{\sum_{i=1}^{d}\|y_i\|_2^2}{1-2\gamma}} = \frac{\|\mathbf{Y}\|_F}{\sqrt{1-2\gamma}} \tag{B.22}$$

On the other hand,

$$\sum_{j=1}^{d} z_j^\top (\mathbf{B}_j - \mathbf{B})\overline{e_j + w_j} = \sum_{j=1}^{d} z_j^\top (e_j + w_j^*)(\overline{e_j + w_j^*} - \overline{e_j + w_j})^\top \overline{e_j + w_j}$$

$$= \sum_{j=1}^{d} (w_j^* - w_j)^\top (\mathbf{I} - \frac{1}{2}\overline{e_j + w_j}\cdot\overline{e_j + w_j}^\top)(e_j + w_j^*)(\overline{e_j + w_j^*} - \overline{e_j + w_j})^\top \overline{e_j + w_j}$$

For any vector x, $\overline{e_j + w_j}\cdot\overline{e_j + w_j}^\top x$ is the projection of x onto the direction $\overline{e_j + w_j}$, so $\frac{1}{2} \leq \|\mathbf{I} - \frac{1}{2}\overline{e_j + w_j}\cdot\overline{e_j + w_j}^\top\|_2 \leq 1$, and

$$|(w_j^* - w_j)^\top (e_j + w_j^*)(\overline{e_j + w_j^*} - \overline{e_j + w_j})^\top \overline{e_j + w_j}| \overset{①}{\leq} |(w_j^* - w_j)^\top (e_j + w_j^*)|\frac{\|w_j^* - w_j\|_2^2}{2(1-2\gamma)}$$

$$\overset{②}{\leq} \frac{\|w_j^* - w_j\|_2^3(1+\gamma)}{2(1-2\gamma)} \leq \frac{\|w_j^* - w_j\|_2^2(1+\gamma)\gamma}{1-2\gamma} \tag{B.23}$$

where ① uses Lemma B.5.1 term 2, and ② uses Cauchy-Schwartz.

Combining (B.21),(B.22),(B.23), we get

$$\sum_{j=1}^{d} z_j^\top \mathbf{B}_j \overline{e_j + w_j} \geq -\frac{(1+\gamma)^2}{(1-\gamma)\sqrt{1-2\gamma}}\|\mathbf{Z}\|_F\|\mathbf{Y}\|_F - \frac{(1+\gamma)\gamma}{1-2\gamma}\|\mathbf{W}^* - \mathbf{W}\|_F^2$$

2. From $\mathbf{C}$ to $\mathbf{C}_j$:

$$\sum_{j=1}^{d} z_j^\top \mathbf{C}\overline{e_j + w_j} = \sum_{i,j} z_j^\top \langle w_i^* - w_i, \overline{e_i + w_i}\rangle \overline{e_i + w_i} \cdot \overline{e_i + w_i}^\top \overline{e_j + w_j}$$

$$\overset{①}{=} \mathrm{Tr}([\mathbf{Z}^\top \mathbf{X}]\left[\overline{\mathbf{I} + \mathbf{W}}^\top \overline{\mathbf{I} + \mathbf{W}}\right]) = \mathrm{Tr}(\mathbf{Z}^\top \mathbf{X}) + \mathrm{Tr}(\mathbf{Z}^\top \mathbf{X}(\overline{\mathbf{I} + \mathbf{W}}^\top \overline{\mathbf{I} + \mathbf{W}} - \mathbf{I}))$$

$$\overset{②}{\geq} \mathrm{Tr}(\mathbf{Z}^\top \mathbf{X}) - \|\mathbf{Z}\|_F \|\mathbf{X}\|_F \|\overline{\mathbf{I} + \mathbf{W}}^\top \overline{\mathbf{I} + \mathbf{W}} - \mathbf{I}\|_2 \overset{③}{\geq} \mathrm{Tr}(\mathbf{Z}^\top \mathbf{X}) - \frac{4\gamma}{(1-\gamma)^2} \|\mathbf{Z}\|_F \|\mathbf{X}\|_F$$

where ① uses Lemma B.9.2 and $x_j = \langle w_j^* - w_j, \overline{e_j + w_j}\rangle \overline{e_j + w_j}$, ② uses $\mathrm{Tr}(\mathbf{A}\mathcal{B}) \geq -\|\mathbf{A}\|_F \|\mathcal{B}\|_F$, and $\|\mathbf{A}\mathcal{B}\|_F \leq \|\mathbf{A}\|_2 \|\mathcal{B}\|_F$, and ③ uses Lemma B.6.1. On the other hand,

$$\sum_{j=1}^{d} z_j^\top (\mathbf{C} - \mathbf{C}_j)\overline{e_j + w_j} = \sum_{j=1}^{d} z_j^\top \langle w_j^* - w_j, \overline{e_j + w_j}\rangle \overline{e_j + w_j} \cdot \overline{e_j + w_j}^\top \overline{e_j + w_j}$$

$$= \sum_{j=1}^{d} z_j^\top \langle w_j^* - w_j, \overline{e_j + w_j}\rangle \overline{e_j + w_j} = \mathrm{Tr}(\mathbf{Z}^\top \mathbf{X})$$

That implies, $\frac{1}{2}\sum_{j=1}^{d} z_j^\top \mathbf{C}_j \overline{e_j + w_j} \geq -\frac{2\gamma}{(1-\gamma)^2} \|\mathbf{Z}\|_F \|\mathbf{X}\|_F$.

3. From $\mathbf{D}$ to $\mathbf{D}_j$:

$$\sum_{j=1}^{d} z_j^\top \mathbf{D}\overline{e_j + w_j} = \sum_{i,j} z_j^\top z_i \overline{e_i + w_i}^\top \overline{e_j + w_j} = \mathrm{Tr}\left([\mathbf{Z}^\top \mathbf{Z}]\left[\overline{\mathbf{I} + \mathbf{W}}^\top \overline{\mathbf{I} + \mathbf{W}}\right]\right) \geq \frac{(1-\gamma)^2}{(1+\gamma)^2} \|\mathbf{Z}\|_F^2$$

where the last inequality holds by Lemma B.6.1. On the other hand,

$$z_j^\top (\mathbf{D} - \mathbf{D}_j)\overline{e_j + w_j} = \|z_j\|_2^2$$

That gives,

$$\sum_j z_j^\top \mathbf{D}_j \overline{e_j + w_j} \geq -\frac{4\gamma}{(1+\gamma)^2} \|\mathbf{Z}\|_F^2$$

Now, combining $\mathbf{B}_j, \mathbf{C}_j, \mathbf{D}_j$ together, using (B.20), we have

$$\sum_{j=1}^{d} z_j^\top \mathbf{A}_j \overline{e_j + w_j} \geq -\frac{(1+\gamma)^2}{(1-\gamma)\sqrt{1-2\gamma}} \|\mathbf{Z}\|_F \|\mathbf{Y}\|_F - \frac{(1+\gamma)\gamma}{1-2\gamma} \|\mathbf{W}^* - \mathbf{W}\|_F^2$$

$$- \frac{2\gamma}{(1-\gamma)^2}\|\mathbf{Z}\|_F\|\mathbf{X}\|_F - \frac{4\gamma}{(1+\gamma)^2}\|\mathbf{Z}\|_F^2$$

By definition, we know $\|\mathbf{X}\|_F \le \|\mathbf{W}^* - \mathbf{W}\|_F, \|\mathbf{Y}\|_F \le \|\mathbf{W}^* - \mathbf{W}\|_F, \|\mathbf{Z}\|_F \le \|\mathbf{W}^* - \mathbf{W}\|_F$, and $\gamma \le \frac{1}{100}$, so

$$- \frac{(1+\gamma)\gamma}{1-2\gamma}\|\mathbf{W}^* - \mathbf{W}\|_F^2 - \frac{2\gamma}{(1-\gamma)^2}\|\mathbf{Z}\|_F\|\mathbf{X}\|_F - \frac{4\gamma}{(1+\gamma)^2}\|\mathbf{Z}\|_F^2 \ge -7\gamma\|\mathbf{W}^* - \mathbf{W}\|_F^2 \tag{B.24}$$

Moreover,

$$-\left(\frac{(1+\gamma)^2}{(1-\gamma)\sqrt{1-2\gamma}} - 1\right)\|\mathbf{Z}\|_F\,\|\mathbf{Y}\|_F \ge -0.05\gamma\|\mathbf{W}^* - \mathbf{W}\|_F^2 \tag{B.25}$$

Thus, those are small order terms. The only term left is $\|\mathbf{Z}\|_F\|\mathbf{Y}\|_F$. By (B.19), we know

$$\|\mathbf{Z}\|_F\|\mathbf{Y}\|_F \le \sqrt{\|\mathbf{W}^* - \mathbf{W}\|_F^2 - \frac{3}{4}\|\mathbf{X}\|_F^2}\sqrt{\|\mathbf{W}^* - \mathbf{W}\|_F^2 - \|\mathbf{X}\|_F^2} \tag{B.26}$$

Combining (B.24), (B.25), (B.26), we get:

$$\sum_{j=1}^{d} z_j^\top \mathbf{A}_j\overline{e_j + w_j} \ge -8\gamma\|\mathbf{W}^* - \mathbf{W}\|_F^2 - \sqrt{\|\mathbf{W}^* - \mathbf{W}\|_f^2 - \frac{3}{4}\|\mathbf{X}\|_F^2}\sqrt{\|\mathbf{W}^* - \mathbf{W}\|_F^2 - \|\mathbf{X}\|_F^2} \qquad \square$$

Now it remains to bound $\sqrt{\|\mathbf{W}^* - \mathbf{W}\|_f^2 - \frac{3}{4}\|\mathbf{X}\|_F^2}\sqrt{\|\mathbf{W}^* - \mathbf{W}\|_F^2 - \|\mathbf{X}\|_F^2}$.

Lemma B.9.4.

$$-\sqrt{\|\mathbf{W}^* - \mathbf{W}\|_F^2 - \frac{3}{4}\|\mathbf{X}\|_F^2}\sqrt{\|\mathbf{W}^* - \mathbf{W}\|_F^2 - \|\mathbf{X}\|_F^2} \ge -1.3\|\mathbf{W}^* - \mathbf{W}\|_F^2 + \|\mathbf{W}^* - \mathbf{W}\|_F\|\mathbf{X}\|_F$$

Proof. Consider the function $f(x) = \sqrt{y^2 - \frac{3}{4}x^2}\sqrt{y^2 - x^2} + xy$, where $x \in [0, y]$. It suffices to show that $f(x) \le 1.3y^2$.

Indeed, we know

$$f'(x) = \frac{x(6x^2 - 7y^2)}{2\sqrt{4y^2 - 3x^2}\sqrt{y^2 - x^2}} + y$$

When $x = 0$, $f'(x) = y > 0$, and when $x \to y$, $f'(x) < 0$. We want to find the place where $f'(x) = 0$, which gives the maximum value. Assume $x = \lambda y$, this is equivalent to solve

$$\lambda y(6(\lambda y)^2 - 7y^2) = -2y\sqrt{4y^2 - 3(\lambda y)^2}\sqrt{y^2 - (\lambda y)^2}$$

Cancel all y, and we get the solution $x \approx 0.566y$, where $f(x) \approx 1.2845y^2 < 1.3y^2$. □

Proof of Lemma B.4.1. Combining Lemma B.9.3 and Lemma B.9.4, we have proved Lemma B.4.1. □

B.9.2 Proof for Lemma B.4.2

Again, we first consider the full sum, $g = \sum_{i=1}^{d}(\|e_i + w_i^*\|_2 - \|e_i + w_i\|_2)$.

By Lemma B.6.3, we have

$$|g - g_j| = |\|e_j + w_j^*\|_2 - \|e_j + w_j\|_2| \le \|w_j^* - w_j\|_2$$

Thus by Cauchy Schwartz,

$$|(g - g_j)\langle w_j^* - w_j, \overline{e_j + w_j}\rangle| \le \|w_j^* - w_j\|_2\|x_j\|_2$$

Summing over j, we get

$$\sum_{j=1}^{d}|(g - g_j)\langle w_j^* - w_j, \overline{e_j + w_j}\rangle| \le \sum_{j=1}^{d}\|w_j^* - w_j\|_2\|x_j\|_2 \le \|\mathbf{W}^* - \mathbf{W}\|_F\|\mathbf{X}\|_F \tag{B.27}$$

where the last inequality is by Cauchy Schwartz.

Now

$$g\sum_{j=1}^{d}\langle w_j^* - w_j, \overline{e_j + w_j}\rangle = g\sum_{j=1}^{d}\langle e_j + w_j^* - e_j + w_j, \overline{e_j + w_j}\rangle$$

$$= g \sum_{j=1}^{d} (\|e_j + w_j^*\|_2 - \|e_j + w_j\|_2 + \langle e_j + w_j^*, \overline{e_j + w_j} - \overline{e_j + w_j^*} \rangle) = g^2 + gb \geq gb \tag{B.28}$$

where b is defined to be $\sum_{j=1}^{d} \langle e_j + w_j^*, \overline{e_j + w_j} - \overline{e_j + w_j^*} \rangle$. By Lemma B.5.1 term 2 we know

$$-\frac{(1+\gamma)\|\mathbf{W}^* - \mathbf{W}\|_F^2}{2(1-2\gamma)} \leq b \leq 0$$

Combining (B.27), (B.28), the lemma follows.

$$\begin{aligned}\sum_{j=1}^{d} \langle g_j \overline{e_j + w_j}, w_j^* - w_j \rangle &= \sum_{j=1}^{d} \langle (g_j - g)\overline{e_j + w_j}, w_j^* - w_j \rangle + \sum_{j=1}^{d} \langle g\overline{e_j + w_j}, w_j^* - w_j \rangle \\ &\geq -\|\mathbf{W}^* - \mathbf{W}\|_F \|\mathbf{X}\|_F + g^2 + gb \geq -\|\mathbf{W}^* - \mathbf{W}\|_F \|\mathbf{X}\|_F - \frac{(1+\gamma)g\|\mathbf{W}^* - \mathbf{W}\|_F^2}{2(1-2\gamma)}\end{aligned}$$

B.9.3 Proof for Lemma B.4.3

$$\begin{aligned}&\sum_{j=1}^{d} \langle \mathbf{P}_{3,j}, w_j^* - w_j \rangle = \sum_{j=1}^{d} \langle \frac{\pi}{2}(w_j^* - w_j) - \theta_{j^*,j}(e_j + w_j^*) + \|e_j + w_j^*\| \sin\theta_{j^*,j} \overline{e_j + w_j}, w_j^* - w_j \rangle \\ \overset{①}{=} &\sum_{j=1}^{d} \langle \frac{\pi}{2}(w_j^* - w_j) - \theta_{j^*,j}\|e_j + w_j^*\|_2 (\overline{e_j + w_j^*} - \overline{e_j + w_j}) + \frac{\alpha_{j^*,j}|\theta_{j^*,j}|^3 \|e_j + w_j^*\| \overline{e_j + w_j}}{3}, w_j^* - w_j \rangle \\ \overset{②}{\geq} &\frac{\pi}{2}\|\mathbf{W}^* - \mathbf{W}\|_F^2 - \sum_{j=1}^{d} 1.001(1+\gamma)\|w_j^* - w_j\|_2^2 \|\overline{e_j + w_j^*} - \overline{e_j + w_j}\|_2 - \sum_{j=1}^{d} 0.335(1+\gamma)\|w_j^* - w_j\|_2^4 \\ \overset{③}{\geq} &\frac{\pi}{2}\|\mathbf{W}^* - \mathbf{W}\|_F^2 - \sum_{j=1}^{d} \frac{1.001(1+\gamma)}{\sqrt{1-2\gamma}}\|w_j^* - w_j\|_2^3 - \sum_{j=1}^{d} 0.335(1+\gamma)\|w_j^* - w_j\|_2^4 \\ \overset{④}{\geq} &\left(\frac{\pi}{2} - 0.021\right)\|\mathbf{W}^* - \mathbf{W}\|_F^2\end{aligned}$$

where ① uses Taylor's Theorem for $\sin\theta_{j^*,j}$, so we know $|\alpha_{j^*,j}| \leq 1$. ② uses Lemma B.5.1 term 3 and Cauchy Schwartz, ③ uses Lemma B.5.1 term 1, ④ holds since $\gamma \leq \frac{1}{100}$, and the two small order terms can be bounded by $0.021\|\mathbf{W}^* - \mathbf{W}\|_F^2$.

APPENDIX C

APPENDIX FOR HARMONICA

C.1 Experimental details

C.1.1 Options

Table C.1: 60 options used in Section 6.5

Option Name	Description
01. Weight initialization	Use standard initializations or other initializations?
02. Weight initialization (Detail 1)	Xavier Glorot [37], Kaiming [50], $1/n$, or $1/n^2$?
03. Optimization method	SGD or ADAM? [72]
04. Initial learning rate	≥ 0.01 or < 0.01?
05. Initial learning rate (Detail 1)	≥ 0.1, < 0.1, ≥ 0.001, or < 0.001?
06. Initial learning rate (Detail 2)	0.3, 0.1, 0.03, 0.01, 0.003, 0.001, 0.0003, or 0.0001?
07. Learning rate drop	Do we need to decrease learning rate as we train? Yes or No?
08. Learning rate first drop time	If drop learning rate, when is the first time to drop by $1/10$? Epoch 40 or Epoch 60?
09. Learning rate second drop time	If drop learning rate, when is the second time to drop by $1/100$? Epoch 80 or Epoch 100?
10. Use momentum [122]	Yes or No?
11. Momentum rate	If use momentum, rate is 0.9 or 0.99?

12. Initial residual link weight	What is the initial residual link weight? All constant 1 or a random number in [0, 1]?
13. Tune residual link weight	Do we want to use back propagation to tune the weight of residual links? Yes or No?
14. Tune time of residual link weight	When do we start to tune residual link weight? At the first epoch or epoch 10?
15. Resblock first activation	Do we want to add activation layer after the first convolution? Yes or No?
16. Resblock second activation	Do we want to add activation layer after the second convolution? Yes or No?
17. Resblock third activation	Do we want to add activation layer after adding the residual link? Yes or No?
18. Convolution bias	Do we want to have bias term in convolutional layers? Yes or No?
19. Activation	What kind of activations do we use? ReLU or others?
20. Activation (Detail 1)	ReLU, ReLU, Sigmoid, or Tanh?
21. Use dropout [119]	Yes or No?
22. Dropout rate	If use dropout, rate is high or low?
23. Dropout rate (Detail 1)	If use dropout, the rate is 0.3, 0.2, 0.1, or 0.05?
24. Batch norm [63]	Do we use batch norm? Yes or No?
25. Batch norm tuning	If we use batch norm, do we tune the parameters in the batch norm layers? Yes or No?
26. Resnet shortcut type	What kind of resnet shortcut type do we use? Identity or others?

27. Resnet shortcut type (Detail 1)	Identity, Identity, Type B or Type C?
28. Weight decay	Do we use weight decay during the training? Yes or No?
29. Weight decay parameter	If use weight decay, what is the parameter? $1e-3$ or $1e-4$?
30. Batch Size	What is the batch size we should use? Big or Small?
31. Batch Size (Detail 1)	256, 128, 64, or 32?
32. Optnet	An option specific to the code[1]. Yes or No?
33. Share gradInput	An option specific to the code. Yes or No?
34. Backend	What kind of backend shall we use? cudnn or cunn?
35. cudnn running state	If use cudnn, shall we use fastest of other states?
36. cudnn running state (Detail 1)	Fastest, Fastest, default, deterministic
37. nthreads	How many threads shall we use? Many or few?
38. nthreads (Detail 1)	8, 4, 2, or 1?
39-60. Dummy variables	Just dummy variables, no effect at all.

See Table C.1 for the specific hyperparameter options that we use in Section 6.5. For those variables with k options ($k > 2$), we use $\log k$ binary variables under the same name to represent them. For example, we have two variables (01, 02) and their binary representation to denote four kinds of possible initializations: Xavier Glorot [37], Kaiming [50], $1/n$, or $1/n^2$.

C.1.2 Importance features

We show the selected important features and their weights during the first 3 stages in Table C.2, where each feature is a monomial of variables with degree at most 3. We do not include the 4th

[1] https://github.com/facebook/fb.resnet.torch

stage because in that stage there are no features with nonzero weights.

Smart choices on important options. Based on Table C.2, Harmonica will fix the following variables (sorted according to their importance): Batch Norm (Yes), Activation (ReLU), Initial learning rate ($[0.001, 0.1]$), Optimization method (Adam), Use momentum (Yes), Resblock first activation (Yes), Resblcok third activation (No), Weight decay (No if initial learning rate is comparatively small and Yes otherwise), Batch norm tuning (Yes). Most of these choices match what people are doing in practice.

A metric for the importance of variables. The features that Harmonica finds can serve as a metric for measuring the importance of different variables. For example, Batch Norm turns out to be the most significant variable, and ReLU is second important. By contrast, Dropout, when Batch Norm is presented, does not have significant contributions. This actually matches with the observations in [63].

No dummy/irrelevant variables selected. Although there are $21/60$ dummy variables, we never select any of them. Moreover, the irrelevant variables like cudnn, backend, nthreads, which do not affect the test error, were not selected.

C.1.3 Generalizing from small networks to big networks

In our experiments, Harmonica first runs on a small network to extract important features and then uses these features to do fine tuning on a big network. Since Harmonica finds significantly better solutions, it is natural to ask whether other algorithms can also exploit this strategy to improve performance.

[2]This is an interesting feature. In the code repository that we use, optnet, shared gradInput are two special options of the code and cannot be set true at the same time, otherwise the training becomes unpredictable.

Table C.2: Important features

Stage	Feature Name	Weights
1-1	24. Batch norm	8.05
1-2	19. Activation	3.47
1-3	04. Initial learning rate * 05. Initial learning rate (Detail 1)	3.12
1-4	19. Activation * 24. Batch norm	-2.55
1-5	04. Initial learning rate	-2.34
1-6	28. Weight decay	-1.90
1-7	24. Batch norm * 28. Weight decay	1.79
1-8	34. Optnet * 35. Share gradInput * 52. Dummy ^{2}	1.54
2-1	03. Optimization method	-4.22
2-2	03. Optimization method * 10. Use momentum	-3.02
2-3	15. Resblock first activation	2.80
2-4	10. Use momentum	2.19
2-5	15. Resblock first activation * 17. Resblock third activation	1.68
2-6	01. Good initialization	-1.26
2-7	01. Good initialization * 10. Use momentum	-1.12
2-8	01. Good initialization * 03. Optimization method	0.67
3-1	29. Weight decay parameter	-0.49
3-2	28. Weight decay	-0.26
3-3	06. Initial learning rate (Detail 3) * 28. Weight decay	0.23
3-4	25. Batch norm tuning	0.21
3-5	28. Weight decay * 29. Weight decay parameter	0.20

Unfortunately, it seems that all the other algorithms do not naturally support feature extraction from a small network. For Bayesian Optimization techniques, small networks and large networks have different optimization spaces. Therefore without some modification, Spearmint cannot use information from the small network to update the prior distribution for the large network.

Random-search-based techniques are able to find configurations with low test error on the small network, which might be good candidates for the large network. However, based on our simulation, good configurations of hyperparameters from random search do not generalize from small networks to large networks. This is in contrast to important features in our (Fourier) space, which do seem to generalize.

To test the latter observation using Cifar-10 dataset, we first spent 7 GPU days on 8 layer network to find top 10 configurations among 300 random selected configurations. Then we apply these 10 configurations, as well as 90 locally perturbed configurations (each of them is obtained by switching one random option from one top-10 configuration), so in total 100 "promising" configurations, to the large 56 layer network. This simulation takes 27 GPU days, but the best test error we obtained is only 11.1%, even worse than purely random search. Since Hyperband is essentially a fast version of Random Search, it also does not support feature extraction.

Hence, being able to extract important features from small networks seems empirically to be a unique feature of Harmonica.

BIBLIOGRAPHY

[1] Alekh Agarwal, Sahand Negahban, and Martin J Wainwright. Fast global convergence rates of gradient methods for high-dimensional statistical recovery. In *Advances in Neural Information Processing Systems*, pages 37–45, 2010.

[2] Zeyuan Allen-Zhu and Yang Yuan. Improved SVRG for non-strongly-convex or sum-of-non-convex objectives. In *ICML 2016*, volume 48, pages 1080–1089, 2016.

[3] Animashree Anandkumar, Rong Ge, Daniel Hsu, Sham M. Kakade, and Matus Telgarsky. Tensor decompositions for learning latent variable models. *Journal of Machine Learning Research*, 15:2773–2832, 2014.

[4] Alexandr Andoni, Rina Panigrahy, Gregory Valiant, and Li Zhang. Learning polynomials with neural networks. In *ICML*, pages 1908–1916, 2014.

[5] Sanjeev Arora, Aditya Bhaskara, Rong Ge, and Tengyu Ma. Provable bounds for learning some deep representations. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pages 584–592, 2014.

[6] Sanjeev Arora, Rong Ge, Ankur Moitra, and Sushant Sachdeva. Provable ICA with unknown gaussian noise, with implications for gaussian mixtures and autoencoders. In *Advances in Neural Information Processing Systems*, pages 2375–2383, 2012.

[7] Kazuoki Azuma. Weighted sums of certain dependent random variables. *Tohoku Mathematical Journal, Second Series*, 19(3):357–367, 1967.

[8] Bowen Baker, Otkrist Gupta, Nikhil Naik, and Ramesh Raskar. Designing neural network architectures using reinforcement learning. *CoRR*, abs/1611.02167, 2016.

[9] Andrew R. Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Trans. Information Theory*, 39(3):930–945, 1993.

[10] Yoshua Bengio. Gradient-based optimization of hyperparameters. *Neural Computation*, 12(8):1889–1900, 2000.

[11] Yoshua Bengio. Learning deep architectures for AI. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.

[12] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *J. Mach. Learn. Res.*, 13:281–305, February 2012.

[13] James S. Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 2546–2554. Curran Associates, Inc., 2011.

[14] Srinadh Bhojanapalli, Behnam Neyshabur, and Nathan Srebro. Global optimality of local search for low rank matrix recovery. In *NIPS 2016*, 2016.

[15] Avrim Blum, Adam Kalai, and Hal Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *J. ACM*, 50(4):506–519, July 2003.

[16] Jean Bourgain. *An Improved Estimate in the Restricted Isometry Problem*, pages 65–70. Springer International Publishing, Cham, 2014.

[17] Leo Breiman. Hinging hyperplanes for regression, classification, and function approximation. *IEEE Trans. Information Theory*, 39(3):999–1013, 1993.

[18] Alon Brutzkus and Amir Globerson. Globally optimal gradient descent for a convnet with gaussian inputs. In *ICML 2017*, 2017.

[19] E. J. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theor.*, 52(2):489–509, February 2006.

[20] J-F Cardoso. Source separation using higher order moments. In *Acoustics, Speech, and Signal Processing*, pages 2109–2112. IEEE, 1989.

[21] P. Chaudhari, A. Choromanska, S. Soatto, Y. LeCun, C. Baldassi, C. Borgs, J. Chayes, L. Sagun, and R. Zecchina. Entropy-SGD: Biasing Gradient Descent Into Wide Valleys. *ArXiv e-prints*, November 2016.

[22] Anna Choromanska, Mikael Henaff, Michaël Mathieu, Gérard Ben Arous, and Yann LeCun. The loss surfaces of multilayer networks. In *AISTATS*, 2015.

[23] Anna Choromanska, Yann LeCun, and Gérard Ben Arous. Open problem: The landscape of the loss surfaces of multilayer networks. In *Proceedings of The 28th Conference on Learning Theory, COLT 2015, Paris, France, July 3-6, 2015*, pages 1756–1760, 2015.

[24] P. Comon. Tensor decompositions. *Mathematics in Signal Processing V*, pages 1–24, 2002.

[25] Pierre Comon, Xavier Luciani, and André LF De Almeida. Tensor decompositions, alternating least squares and other tales. *Journal of Chemometrics*, 23(7-8):393–405, 2009.

[26] George Cybenko. Approximation by superpositions of a sigmoidal function. *MCSS*, 5(4):455, 1992.

[27] Yann N Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In *Advances in Neural Information Processing Systems*, pages 2933–2941, 2014.

[28] Aaron Defazio, Francis R. Bach, and Simon Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *NIPS 2014*, pages 1646–1654, 2014.

[29] L. Dinh, R. Pascanu, S. Bengio, and Y. Bengio. Sharp Minima Can Generalize For Deep Nets. *ArXiv e-prints*, March 2017.

[30] D. L. Donoho. Compressed sensing. *IEEE Trans. Inf. Theor.*, 52(4):1289–1306, April 2006.

[31] V. Feldman, Parikshit Gopalan, Subhash Khot, and Ashok Kumar Ponnuswami. On agnostic learning parities, monomials,and halfspaces. *SIAM Journal on Computing*, 39(2):606–645, 2009.

[32] Alan Frieze, Mark Jerrum, and Ravi Kannan. Learning linear transformations. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 359–359, 1996.

[33] Jie Fu, Hongyin Luo, Jiashi Feng, Kian Hsiang Low, and Tat-Seng Chua. Drmad: Distilling reverse-mode automatic differentiation for optimizing hyperparameters of deep neural networks. *CoRR*, abs/1601.00917, 2016.

[34] Jacob R. Gardner, Matt J. Kusner, Zhixiang Eddie Xu, Kilian Q. Weinberger, and John P. Cunningham. Bayesian optimization with inequality constraints. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pages 937–945, 2014.

[35] Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points - online stochastic gradient for tensor decomposition. In *COLT 2015*, volume 40, pages 797–842, 2015.

[36] Rong Ge, Jason D. Lee, and Tengyu Ma. Matrix completion has no spurious local minimum. In *NIPS 2016*, 2016.

[37] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*, pages 249–256, 2010.

[38] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *AISTATS*, pages 315–323, 2011.

[39] Surbhi Goel, Varun Kanade, Adam R. Klivans, and Justin Thaler. Reliably learning the relu in polynomial time. *CoRR*, abs/1611.10258, 2016.

[40] Surbhi Goel and Adam Klivans. Eigenvalue decay implies polynomial-time learnability for neural networks. In *NIPS 2017*, 2017.

[41] Surbhi Goel and Adam Klivans. Learning Depth-Three Neural Networks in Polynomial Time. *ArXiv e-prints*, 2017.

[42] Ian J. Goodfellow and Oriol Vinyals. Qualitatively characterizing neural network optimization problems. *CoRR*, abs/1412.6544, 2014.

[43] Morgan A Hanson. Invexity and the kuhn–tucker theorem. *Journal of mathematical analysis and applications*, 236(2):594–604, 1999.

[44] M. Hardt, B. Recht, and Y. Singer. Train faster, generalize better: Stability of stochastic gradient descent. *ArXiv e-prints*, September 2015.

[45] Moritz Hardt and Tengyu Ma. Identity matters in deep learning. *CoRR*, abs/1611.04231, 2016.

[46] Richard A Harshman. Foundations of the PARAFAC procedure: Models and conditions for an “explanatory" multi-modal factor analysis. *UCLA Working Papers in Phonetics*, 16(1):84, 1970.

[47] Ishay Haviv and Oded Regev. The list-decoding size of fourier-sparse boolean functions. In David Zuckerman, editor, *30th Conference on Computational Complexity, CCC 2015, June 17-19, 2015, Portland, Oregon, USA*, volume 33 of *LIPIcs*, pages 58–71. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015.

[48] Ishay Haviv and Oded Regev. The restricted isometry property of subsampled fourier matrices. In *Proceedings of the Twenty-seventh Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '16, pages 288–297, Philadelphia, PA, USA, 2016. Society for Industrial and Applied Mathematics.

[49] Elad Hazan, Adam R. Klivans, and Yang Yuan. Hyperparameter optimization: A spectral approach. *CoRR*, abs/1706.00764, 2017.

[50] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 1026–1034, 2015.

[51] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[52] Sepp Hochreiter and JÃijrgen Schmidhuber. Simplifying neural nets by discovering flat minima. In *Advances in Neural Information Processing Systems 7*, pages 529–536. MIT Press, 1995.

[53] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

[54] Elad Hoffer, Itay Hubara, and Daniel Soudry. Train longer, generalize better: closing the generalization gap in large batch training of neural networks. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 1729–1739. Curran Associates, Inc., 2017.

[55] Kurt Hornik, Maxwell B. Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.

[56] Furong Huang, UN Niranjan, Mohammad Umar Hakeem, and Animashree Anandkumar. Fast detection of overlapping communities via online tensor methods. *arXiv:1309.0787*, 2013.

[57] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten. Densely Connected Convolutional Networks. *ArXiv e-prints*, August 2016.

[58] Gao Huang, Yixuan Li, Geoff Pleiss, Zhuang Liu, John E. Hopcroft, and Kilian Q. Weinberger. Snapshot ensembles: Train 1, get m for free. In *ICLR 2017*, 2017.

[59] Aapo Hyvarinen. Fast ICA for noisy data using gaussian moments. In *Circuits and Systems*, volume 5, pages 57–61, 1999.

[60] Aapo Hyvärinen, Juha Karhunen, and Erkki Oja. *Independent component analysis*, volume 46. John Wiley & Sons, 2004.

[61] Ilija Ilievski, Taimoor Akhtar, Jiashi Feng, and Christine Annette Shoemaker. Efficient hyperparameter optimization for deep learning algorithms using deterministic RBF surrogates. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, pages 822–829, 2017.

[62] Masato Inoue, Hyeyoung Park, and Masato Okada. On-line learning theory of soft committee machines with correlated hidden units–steepest gradient descent and natural gradient descent–. *Journal of the Physical Society of Japan*, 72(4):805–810, 2003.

[63] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 448–456, 2015.

[64] Prateek Jain, Praneeth Netrapalli, and Sujay Sanghavi. Low-rank matrix completion using

alternating minimization. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 665–674, 2013.

[65] Kevin G. Jamieson and Ameet Talwalkar. Non-stochastic best arm identification and hyperparameter optimization. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, AISTATS 2016, Cadiz, Spain, May 9-11, 2016*, pages 240–248, 2016.

[66] Majid Janzamin, Hanie Sedghi, and Anima Anandkumar. Beating the perils of non-convexity: Guaranteed training of neural networks using tensor methods. *arXiv preprint arXiv:1506.08473*, 2015.

[67] C. Jin, P. Netrapalli, and M. I. Jordan. Accelerated Gradient Descent Escapes Saddle Points Faster than Gradient Descent. *ArXiv e-prints*, November 2017.

[68] Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M. Kakade, and Michael I. Jordan. How to escape saddle points efficiently. *CoRR*, abs/1703.00887, 2017.

[69] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *NIPS 2013*, pages 315–323, 2013.

[70] Kenji Kawaguchi. Deep learning without poor local minima. In *NIPS*, pages 586–594, 2016.

[71] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *ICLR 2017*, 2017.

[72] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[73] Krzysztof C Kiwiel. Convergence and efficiency of subgradient methods for quasiconvex minimization. *Mathematical programming*, 90(1):1–25, 2001.

[74] R. Kleinberg, Y. Li, and Y. Yuan. An Alternative View: When Does SGD Escape Local Minima? *ArXiv e-prints*, February 2018.

[75] J. M. Klusowski and A. R. Barron. Risk Bounds for High-dimensional Ridge Function Combinations Including Neural Networks. *ArXiv e-prints*, July 2016.

[76] Murat Kocaoglu, Karthikeyan Shanmugam, Alexandros G. Dimakis, and Adam R. Klivans. Sparse polynomial learning and graph sketching. In Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3122–3130, 2014.

[77] Tamara G Kolda. Orthogonal tensor decompositions. *SIAM Journal on Matrix Analysis and Applications*, 23(1):243–255, 2001.

[78] Yann LeCun, Leon Bottou, Genevieve B. Orr, and Klaus Robert Müller. *Efficient BackProp*, pages 9–50. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.

[79] J. D. Lee, M. Simchowitz, M. I. Jordan, and B. Recht. Gradient Descent Converges to Minimizers. *ArXiv e-prints*, February 2016.

[80] Kfir Y. Levy. The power of normalization: Faster evasion of saddle points. *CoRR*, abs/1611.04831, 2016.

[81] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar. Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization. *ArXiv e-prints*, March 2016.

[82] Yuanzhi Li and Yang Yuan. Convergence analysis of two-layer neural networks with relu activation. In *NIPS 2017*, 2017.

[83] Nathan Linial, Yishay Mansour, and Noam Nisan. Constant depth circuits, fourier transform, and learnability. *J. ACM*, 40(3):607–620, July 1993.

[84] Roi Livni, Shai Shalev-Shwartz, and Ohad Shamir. On the computational efficiency of training neural networks. In *NIPS*, pages 855–863, 2014.

[85] Ilya Loshchilov and Frank Hutter. SGDR: stochastic gradient descent with restarts. In *ICLR 2017*, 2017.

[86] Jelena Luketina, Mathias Berglund, Klaus Greff, and Tapani Raiko. Scalable gradient-based tuning of continuous regularization hyperparameters. *CoRR*, abs/1511.06727, 2015.

[87] Dougal Maclaurin, David Duvenaud, and Ryan P. Adams. Gradient-based hyperparameter optimization through reversible learning. In *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pages 2113–2122. JMLR.org, 2015.

[88] S. Mandt, M. D. Hoffman, and D. M. Blei. Stochastic Gradient Descent as Approximate Bayesian Inference. *ArXiv e-prints*, April 2017.

[89] Olvi L Mangasarian. Pseudo-convex functions. *Journal of the Society for Industrial & Applied Mathematics, Series A: Control*, 3(2):281–290, 1965.

[90] Yishay Mansour. *Learning Boolean Functions via the Fourier Transform*, pages 391–424. Springer US, Boston, MA, 1994.

[91] Guido F. Montúfar, Razvan Pascanu, KyungHyun Cho, and Yoshua Bengio. On the number of linear regions of deep neural networks. In *NIPS*, pages 2924–2932, 2014.

[92] W. Mou, L. Wang, X. Zhai, and K. Zheng. Generalization Bounds of SGLD for Non-convex Learning: Two Theoretical Viewpoints. *ArXiv e-prints*, July 2017.

[93] Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, pages 807–814, 2010.

[94] Sahand Negahban and Devavrat Shah. Learning sparse boolean polynomials. In *Allerton*, pages 2032–2036. IEEE, 2012.

[95] Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer Publishing Company, Incorporated, 1 edition, 2014.

[96] Ryan O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, New York, NY, USA, 2014.

[97] Bruno A Olshausen and David J Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision research*, 37(23):3311–3325, 1997.

[98] Xingyuan Pan and Vivek Srikumar. Expressiveness of rectifier networks. In *ICML*, pages 2427–2435, 2016.

[99] Razvan Pascanu, Guido Montúfar, and Yoshua Bengio. On the number of inference regions of deep feed forward networks with piece-wise linear activations. *CoRR*, abs/1312.6098, 2013.

[100] V. Patel. The Impact of Local Geometry and Batch Size on the Convergence and Divergence of Stochastic Gradient Descent. *ArXiv e-prints*, September 2017.

[101] Alexander Rakhlin, Ohad Shamir, and Karthik Sridharan. Making gradient descent optimal for strongly convex stochastic optimization. In *ICML*, pages 449–456, 2012.

[102] Magnus Rattray, David Saad, and Shun-ichi Amari. Natural gradient descent for on-line learning. *Physical review letters*, 81(24):5461, 1998.

[103] Holger Rauhut. Compressive sensing and structured random matrices. *Theoretical foundations and numerical methods for sparse recovery*, 9:1–92, 2010.

[104] Benjamin Recht. Embracing the random. http://www.argmin.net/2016/06/23/hyperband/, 2016.

[105] Benjamin Recht. The news on auto-tuning. http://www.argmin.net/2016/06/20/hypertuning/, 2016.

[106] M. Rudelson and R. Vershynin. Non-asymptotic theory of random matrices: extreme singular values. *ArXiv e-prints*, 2010.

[107] David Saad and Sara A Solla. On-line learning in soft committee machines. *Physical Review E*, 52(4):4225, 1995.

[108] David Saad and Sara A. Solla. Dynamics of on-line gradient descent learning for multilayer neural networks. *Advances in Neural Information Processing Systems*, 8:302–308, 1996.

[109] I. Safran and O. Shamir. Spurious Local Minima are Common in Two-Layer ReLU Neural Networks. *ArXiv e-prints*, December 2017.

[110] Itay Safran and Ohad Shamir. On the quality of the initial basin in overspecified neural networks. In *ICML*, pages 774–782, 2016.

[111] Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *CoRR*, abs/1312.6120, 2013.

[112] Mark Schmidt, Nicolas Le Roux, and Francis Bach. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, pages 1–30, 2016.

[113] Hanie Sedghi and Anima Anandkumar. Provable methods for training neural networks with sparse connectivity. *ICLR*, 2015.

[114] Ohad Shamir. Distribution-specific hardness of learning neural networks. *CoRR*, abs/1609.01037, 2016.

[115] Jirí Síma. Training a single sigmoidal neuron is hard. *Neural Computation*, 14(11):2709–2728, 2002.

[116] S. L. Smith and Q. V. Le. A Bayesian Perspective on Generalization and Stochastic Gradient Descent. *ArXiv e-prints*, October 2017.

[117] Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States.*, pages 2960–2968, 2012.

[118] Jasper Snoek, Kevin Swersky, Richard S. Zemel, and Ryan P. Adams. Input warping for bayesian optimization of non-stationary functions. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pages 1674–1682, 2014.

[119] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[120] Peter Stobbe and Andreas Krause. Learning fourier sparse set functions. In Neil D. Lawrence and Mark A. Girolami, editors, *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2012, La Palma, Canary Islands, April 21-23, 2012*, volume 22 of *JMLR Proceedings*, pages 1125–1133. JMLR.org, 2012.

[121] Ju Sun, Qing Qu, and John Wright. A geometric analysis of phase retrieval. In *IEEE International Symposium on Information Theory, ISIT 2016, Barcelona, Spain, July 10-15, 2016*, pages 2379–2383, 2016.

[122] Ilya Sutskever, James Martens, George E. Dahl, and Geoffrey E. Hinton. On the importance of initialization and momentum in deep learning. In *ICML*, pages 1139–1147, 2013.

[123] Kevin Swersky, Jasper Snoek, and Ryan Prescott Adams. Multi-task bayesian optimization. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 2004–2012, 2013.

[124] Yuandong Tian. Symmetry-breaking convergence analysis of certain two-layered neural networks with relu nonlinearity. In *Submitted to ICLR 2017*, 2016.

[125] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society (Series B)*, 58:267–288, 1996.

[126] Ziyu Wang, Masrour Zoghi, Frank Hutter, David Matheson, and Nando de Freitas. Bayesian optimization in high dimensions via random embeddings. In *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013*, pages 1778–1784, 2013.

[127] Stephen J Wright and Jorge Nocedal. *Numerical optimization*, volume 2. Springer New York, 1999.

[128] Jian Wu and Peter I. Frazier. The parallel knowledge gradient method for batch bayesian optimization. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 3126–3134, 2016.

[129] Bo Xie, Yingyu Liang, and Le Song. Diversity leads to generalization in neural networks. In *AISTATS*, 2017.

[130] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *CoRR*, abs/1605.07146, 2016.

[131] Yuchen Zhang, Jason D. Lee, Martin J. Wainwright, and Michael I. Jordan. Learning halfspaces and neural networks with random initialization. *CoRR*, abs/1511.07948, 2015.

[132] Yuchen Zhang, Percy Liang, and Moses Charikar. A hitting time analysis of stochastic gradient langevin dynamics. *CoRR*, abs/1702.05575, 2017.

[133] Kai Zhong, Zhao Song, Prateek Jain, Peter L. Bartlett, and Inderjit S. Dhillon. Recovery guarantees for one-hidden-layer neural networks. In *ICML 2017*, 2017.

[134] Zhao Zhong, Junjie Yan, and Cheng-Lin Liu. Practical network blocks design with q-learning. *CoRR*, abs/1708.05552, 2017.

[135] Barret Zoph and Quoc V. Le. Neural architecture search with reinforcement learning. *CoRR*, abs/1611.01578, 2016.

[136] J. Y. Zou, D. Hsu, D. C. Parkes, and R. P. Adams. Contrastive learning using spectral methods. In *Advances in Neural Information Processing Systems*, pages 2238–2246, 2013.